

O autor deste livro, associando uma experiência de ensino a um profundo conhecimento da informática, resolveu elaborar este completíssimo manual, orientando-o para diferentes aplicações da informática na escola, abrangendo desde aspectos de organização até aos de carácter pedagógico. Na primeira parte do livro, são abordados todos os problemas que se prendem com a escolha do sistema, a constituição de uma secção de informática na escola, os cuidados a ter com os diversos componentes, sendo ainda levantada a questão de uma rede informática inter-escolas. Na segunda parte da obra, são propostos métodos de trabalho, com vista às diversas aplicações práticas, com exemplos de programas para disciplinas como a Geografia, as Matemáticas, os Trabalhos Oficiais e outras, ou ainda, na avaliação de conhecimentos e na elaboração de ficheiros de alunos. Os Anexos, no final do livro, incluem uma extensa listagem de programas, uma lista de obras a consultar e um glossário de termos de informática. Mas estes são apenas alguns dos assuntos aqui tratados... Eis aqui uma edição muito rica e muito útil, não só pelo seu conteúdo concreto, mas também pelo mérito que tem, de lançar pistas para a exploração das ilimitadas potencialidades de pequenas maravilhas da técnica, ao serviço do ensino e da formação de jovens.

COLECÇÃO SISTEMAS

1. A INFORMÁTICA NA ESCOLA
Manual de Utilização do ZX Spectrum
(e tc 2068), *Luís de Campos*

EDITORIAL  PRESENÇA

A INFORMÁTICA NA ESCOLA
MANUAL DE UTILIZAÇÃO DO ZX SPECTRUM (e tc 2068)



A INFORMÁTICA NA ESCOLA

MANUAL DE UTILIZAÇÃO
DO ZX SPECTRUM (e tc 2068)

LUÍS DE CAMPOS

COLECÇÃO
SISTEMAS



PRESENÇA

A Informática na Escola (Preparatória e Secundária)

**MANUAL DE UTILIZAÇÃO
DO ZX SPECTRUM (E TC 2068)**
(com listagens para o TS 1500)

Luís de Campos

A Informática na Escola

(Preparatória e Secundária)

**MANUAL DE UTILIZAÇÃO
DO ZX SPECTRUM (E TC 2068)
(com listagens para o TS 1500)**

EDITORIAL  PRESENÇA

De nós — eu e a Fátima — nasceram já muitas coisas. Geralmente boas, e muitas vezes belas. A ideia de escrever este livro foi uma dessas coisas. Por isso — e pelo conteúdo — é dela e para ela também esta obra.

Para o IVO — para que a escola do seu tempo seja diferente e ele possa ser actuante nessa escola — é também muito do que faço, neste caso o livro.

Livro que teria tido, por certo, alguma dificuldade em criar (ao menos de forma tão acabada) sem a imprescindível ajuda do Quim (com a sua boa vontade e empenho nos desenhos) e do B.B. (que me aturou a teimosia durante a revisão que tão bem fez). E que foi indubitavelmente valorizado pelos dois programas cedidos pelo Catalão e o José Augusto, bem como pelos desenhos do Almeida.

Quero ainda dedicá-lo, muito especialmente, aos meus colegas e alunos, não podendo deixar de destacar, com uma particular amizade, o Vasco.

PREÂMBULO

Em fins de 1984, o Estado decidiu iniciar um processo de informatização das escolas. E não só.

Assim, a par da criação da Inforjovem — sector destinado a levar conhecimentos de informática à juventude nos diversos pontos do país, e que montou centros regionais de computadores com cursos de Basic e possibilidades de utilização —, o Ministério da Educação adquiriu também computadores para as escolas do Ensino Preparatório e Secundário.

Por razões de algum modo compreensíveis, algumas escolas reagiram negativamente ou passivamente a tais iniciativas — o que não impediu, nem podia impedir, a entrada do computador nas mesmas.

Em muitas delas...

Sozinho, casualmente debaixo de um braço humano — demasiado humano —, o computador entrou, sem que quase ninguém desse por isso.

Muito rapidamente — fruto de um receio incompreensível — foi colocado num cofre, desligado e frio, no escuro que o ruído de uma pesada porta de aço selou.

Alunos e alguns professores — ignorando ainda essa nova existência naquela escola — continuaram a sonhar com a maravilha que seria utilizar computadores nas aulas. E com os resultados positivos que daí adviriam para uns e outros.

Escondido, distante, o computador permanecia tranquilo e selado no meio de amarelados dossiers inúteis — tão inúteis como ele —, naquele momento constituindo apenas um conjunto de matéria transformada, mas morta.

Os que sabiam como ligá-lo e utilizá-lo, criando-lhe a vida obediente de um poderoso auxiliar, desconheciam a sua existência.

Os que sabiam da sua existência... tinham receio.

E o computador continuou no cofre... durante meses...

Um dia... os que sabiam utilizá-lo souberam também que ele existia. Foram buscá-lo, ligaram-no... e mostraram aos outros algo do que ele poderia fazer... como poderia *ajudar* a solucionar facilmente alguns problemas até aí complexos.

E os outros sorriram, também...

E vieram mais computadores... ligados à corrente e aos professores e alunos... ligando alunos a professores... ajudando.
Hoje... aquela escola é diferente.

★

Para que haja cada vez mais escolas diferentes...
Para que o receio de alguns possa morrer em segredo, talvez em casa e sem necessidade de mostrar desconhecimento...
Para que a aprendizagem e o diálogo se tornem mais fáceis, entre todos...
Escrevi este livro.

I PARTE

MONTAGEM DE UM SISTEMA INFORMÁTICO

A ESCOLHA DO SISTEMA

- 1.1 — O que é um sistema informático
- 1.2 — Os elementos do sistema
- 1.3 — O sistema (eventualmente) existente
- 1.4 — O sistema possível/aconselhável

1.1 — O que é um sistema informático

Quando se fala habitualmente em adquirir um computador diz-se apenas isso e, frequentemente e por falta de conhecimento, desconhece-se este facto básico e simples: *um computador sem mais nada, de nada serve!*

Ou seja: o computador é, de facto, uma máquina poderosa e com capacidades de trabalho muito diversificadas; tanto que é quase verdade o poder dizer-se que não há limites para aquilo que um computador pode fazer. Todavia, essa máquina encontra-se encerrada dentro de uma caixa que traz apenas um processo de comunicação com o exterior (o teclado), e mesmo esse destinado a *receber* informação ou ordens.

O ser humano comunica através de sentidos (visão, audição, etc.) que lhe permitem *receber* ou *emitir* informação. Pode recebê-la através da *visão* (imagem) ou *audição* (som) — para o objectivo da presente explicação ignoraremos os restantes sentidos —; pode emití-la através da *voz* (som) ou utilizando as mãos e instrumentos intermédios (canetas, lápis, pincéis, cores, teclados, etc.) e códigos sistematizados (a linguagem escrita, oral, visual, etc.).



Fig. 1

Temos assim dois sistemas potenciais emissores/receptores de informação: o ser humano e o computador, tendo este segundo sido desenvolvido pelo primeiro, para o servir. Contudo, e pelo que atrás ficou dito, estes dois sistemas utilizam *canais* diferentes para emitir ou receber informação codificada também diferentemente: o ser humano utiliza o *ar* (tanto para o som como para a imagem); o computador utiliza *placas* ou *fios* minerais. O primeiro fá-lo através do código linguístico (sinais gráficos correspondentes a sons); o segundo através de um código eléctrico (variações de correntes).

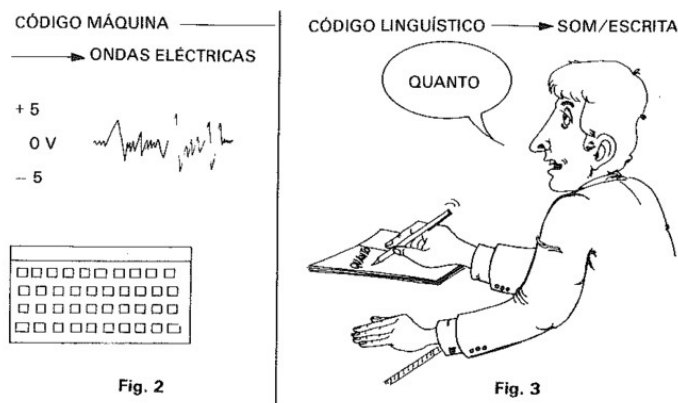


Fig. 2

Fig. 3

Pôr estes dois sistemas em correspondência biunívoca (nos dois sentidos) implica, pois, encontrar *meios transformadores* da informação que cada um *recebe e emite*, de tal modo que cada um dos sistemas possa emitir no *seu código*, recebendo igualmente no *seu código*, toda a informação permutada entre ambos.



Fig. 4

O computador vem já equipado com o seu «transformador» do código humano: o teclado. O ser humano terá de interpor, entre si e o computador, um meio visual (o écran) que lhe permita receber claramente a informação do computador.

Por isso se deve falar em sistema, já que o conjunto mínimo necessário para trabalhar com um computador se compõe precisamente do computador e de um monitor de televisão. Sem estes dois elementos não há comunicação.

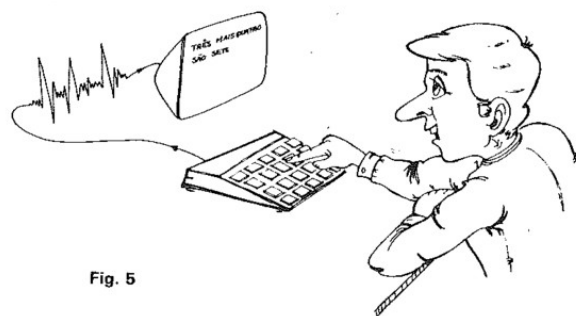


Fig. 5

Mas se este é, teoricamente, o sistema mínimo, na prática é necessário ainda mais um elemento: o *gravador*. E é fácil entender porquê.

O computador é uma ferramenta com memória capaz de guardar e modificar informação fornecida pelo ser humano, apresentando resultados dessa modificação. Contrariamente ao ser humano, o computador não está sempre em funcionamento; para que tal aconteça tem de ter energia eléctrica a circular nos seus circuitos electrónicos. Quando esta é cortada o computador fica virtualmente «morto» — facto que limpa inteiramente as suas memórias de tudo quanto ali tinha sido registado. Desse modo, é conveniente (mesmo indispensável) recorrer a um qualquer processo de *guardar* a informação básica e transformada (ou pelo menos esta) para posterior utilização ou desenvolvimento. O meio mais económico e simples encontrado foi precisamente o gravador de cassettes.

Antes de se desligar o computador, guarda-se, pois, em fita magnética, toda a informação contida nas suas memórias. Quando se pretende retomar o trabalho basta transferir essa informação no sentido inverso — do gravador para as memórias do computador.

Em conclusão, um sistema informático é um conjunto constituído no mínimo por três elementos: o *computador* — elemento central do sistema e que executa todo o trabalho de modificação de informação; o *écran* (televisor ou monitor) — elemento destinado a estabelecer contacto entre o computador e o ser humano; e o *gravador* — memória auxiliar e infinita do computador (e, claro, armazém do produto de trabalho do ser humano).

1.2 — Os elementos do sistema

Como atrás disse, o conjunto dos três elementos constitui apenas o sistema mínimo possível para que o ser humano possa trabalhar utilmente com um computador. Todavia, um sistema completo tem muito mais potencialidades, podendo, portanto, executar um número e diversidade de tarefas muito maior.

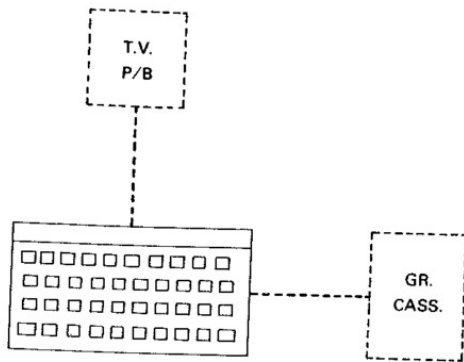


Fig. 6

Continuando ainda a estabelecer um certo paralelismo com o ser humano, temos que este é igualmente uma «máquina» complexa e com várias capacidades (inclusive a de, por exemplo, conceber e desenvolver instrumentos tão elaborados e evoluídos como um computador!). Dando-se conta, progressivamente, dessas capacidades e das possibilidades que as mesmas permitiam, o ser humano foi desenvolvendo extensões dessas capacidades, com os objectivos mais distintos.

No âmbito da imagem, por exemplo, descobriu e fabricou substâncias que lhe permitissem utilizar a cor na criação de pinturas e desenhos, pincéis e outras ferramentas; a máquina fotográfica e a cinematográfica; a câmara de vídeo, a película e a fita magnética — tudo isto para registar o produto do seu trabalho com a imagem.

No capítulo do som, inventou e desenvolveu instrumentos musicais; microfones destinados à captação desse elemento de comunicação; amplificadores e outros equipamentos destinados a manipular o som; discos e fitas magnéticas, para o seu registo e posterior utilização.

Como extensão da memória, começou pelo papel que lhe permite guardar (e transmitir) informação de forma simples e segura; inventou depois a película a que já nos referimos; a fita magnética; etc.

Seria fastidioso focar tudo aquilo que o ser humano desenvolveu com o objectivo de amplificar as suas potencialidades, mas penso que os exemplos dados constituem já uma base suficiente para entender o que vai seguir-se.

Sendo o computador, como atrás disse, uma máquina poderosa e com muitas potencialidades, o ser humano foi igualmente desenvolvendo extensões destinadas a amplificar as suas potencialidades. São aquilo que em informática se designa por *periféricos*. Em conjunto com o computador, constituem, pois, um sistema informático.

Façamos, então, uma breve descrição dos elementos que compõem um sistema informático completo.

O computador

É a unidade central do sistema, sem a qual este não tem razão de existir.

O «cérebro» do computador é, por sua vez, a Unidade Central de Processamento (CPU). Mais pequeno que uma caixa de fósforos, é um circuito integrado que realiza *todos* os cálculos do computador, funcionando com o seu próprio código⁽¹⁾.

Para fazer funcionar o cérebro é necessário que lhe cheguem ordens. E as ordens são comunicadas e recebidas em códigos (linguagens e outros). Portanto — e para que o computador fosse uma máquina universal — houve que inventar uma linguagem intermédia entre o ser humano e o computador: o BASIC (básico). (Desenvolveram-se posteriormente outras, mas esta é a mais divulgada e simples). Ora, criado um código, foi necessário sistematizá-lo numa «gramática», com regras de funcionamento e utilização (sintaxe). Partindo embora de uma mesma estrutura, cada fabricante «gramaticalizou» o seu BASIC, estabelecendo, a partir do mesmo, um conjunto de palavras-chave destinadas a dar determinadas ordens ao computador. Esse conjunto de palavras-chave, bem como as possíveis combinações entre elas, é registado *definitivamente* num outro circuito integrado (a ROM) que constitui aquilo a que se chama a *memória residente do computador*. Desligado ou ligado, esse conjunto de instruções está sempre presente e o ser humano não pode alterar essa memória.

São, assim, necessários outros tipos de memória — aqueles que permitam, consoante as necessidades do utilizador, guardar ou ir buscar informação. Trata-se de um número variável de circuitos integrados de acesso aleatório (RAM), que, no seu conjunto, possuem capacidades de memorização de informação diversas (dependendo das capacidades do computador). Cada unidade de informação é chamada de *byte* e os computadores de uso pessoal possuem entre 16K e 64K (16 000 e 64 000) de memória RAM.

Por fim (e para não tornar esta primeira explicação complexa em demasia), o computador possui «portas» de entrada e saída de informação, através das quais pode ser ligado a extensões (*periféricos*). O mais

⁽¹⁾ O código do microprocessador é o binário. Funciona apenas com 0 e 1. Zero (0) é sinal de desligado; um (1) é sinal de ligado. Das múltiplas possibilidades de combinação de «zeros» e «uns» num conjunto de 8 bits (1 byte) resultam os caracteres e as palavras-chave.

importante destes periféricos (e que equipa, de fábrica, o computador) é o teclado. Semelhante ao de uma máquina de escrever, possui no entanto, e para além das letras, as tais palavras-chave que permitem comandar o trabalho do computador.

Os periféricos

A) De saída de informação

Écran — já anteriormente nos referimos a este periférico. Destina-se exclusivamente a enviar informação visual para o ser humano e é constituído basicamente por um cinescópio (écran electrónico de televisão);

Impressora — sendo o écran um meio de saída de informação visual que, todavia, a não regista de forma permanente, a impressora tem esta última função: registar em papel toda a informação que se pretenda, proveniente do computador;

Sintetizador de som — destina-se a transformar os elementos sonoros (fala, ruídos e música) produzidos pelo computador em ondas capazes de serem entendidas pelo ouvido humano;

Plotter — destina-se a transformar em traços, desenhos, etc., toda a informação gráfica produzida pelo computador (há impressoras que podem, também, executar parcialmente este trabalho).

B) De entrada de informação

Teclado — já acima referido e explicado;

Mesa digitalizadora — tem a função inversa do plotter: transforma em código binário qualquer desenho humano;

Caneta óptica (light pen) — permite realizar desenhos no écran, transformando-os simultaneamente em código binário e fazendo um retorno ao écran;

Sintetizador de som — tem o papel inverso do descrito na alínea anterior: transforma a voz humana em código binário reconhecido pelo computador;

C) De saída e entrada de informação

Modem — dispositivo que permite estabelecer ligação entre computadores, via linha telefónica normal, para troca de informação e outras funções.

D) Extensões de memória

Tratando-se igualmente de periféricos de saída/entrada de informação, deverão ser tratados à parte.

Gravador (cassetes ou fita) — é o periférico mais acessível (economicamente) para guardar programas ou dados e, por isso, o mais utilizado. Tendo uma capacidade infinita (depende do

número de cassetes ou bobinas que se utilize), é todavia muito lento para determinado tipo de trabalhos, sendo também limitado na quantidade de informação que pode guardar de cada vez (apenas a existente na memória do computador);

Discos ou diskettes — sendo o mais caro, é também o periférico ideal para acesso rápido a elevadas quantidades de dados. Atendendo à rapidez e segurança de funcionamento, pode considerar-se na prática como uma verdadeira extensão de memória de acesso imediato, o que transforma qualquer pequeno computador numa máquina poderosíssima;

Microdrives — desenvolvidas inicialmente pela Sinclair Research Corporation, estão já a ser utilizadas em outros computadores (BBC, IBM, etc.). Trata-se de um periférico intermédio entre o gravador de cassetes e a diskette. Utilizando umas pequenas cassetes sem fim, com uma capacidade da ordem dos 90K (para o Spectrum), permite um acesso muito mais rápido aos blocos de informação (15 segundos para um programa de 4 minutos em cassette).

Além destes, há uma série de periféricos, alguns mesmo ainda não perfeitamente conhecidos (robots, etc.). Todavia, e atendendo aos objectivos desta obra, parecem-me ser estes os mais importantes.

1.3 — O sistema (eventualmente) existente

O TS 1500

Por razões ignoradas⁽¹⁾, o Ministério da Educação «impôs» às escolas o TS 1500.

Trata-se da versão Timex do ZX 81 da Sinclair e do TK 85 da Microdigital brasileira, equipamentos que serviriam basicamente para o desenvolvimento do ZX Spectrum, TS 2068 e TC 2068. Sendo, embora, de preço razoavelmente inferior ao Spectrum (e ao TC 2068), é um computador muitíssimo limitado (o máximo de memória possível, adquirindo suplementarmente uma placa expansora, são 16 Kilobytes), lento e desactualizado (o seu ano de fabrico inicial foi 1981).

De qualquer modo, é o que muitas escolas têm neste momento e, como tal, importa apontar o sistema correspondente.

⁽¹⁾ Digo ignoradas porque me parece possível ter-se tratado de pura incompetência por parte do funcionário (ou equipa) responsável por tal decisão. Já uma vez, aliás, um funcionário superior do ME afirmou não ser necessária a cor para o ensino, porque tanto fazia ver o sangue a vermelho como a cinzento (sic). Terá sido este tipo de raciocínio (aliado ao preço) que levou à decisão de compra? De qualquer modo, foi uma maneira de condenar as Escolas a mais um atraso cultural, ao invés de «dar uma ajuda» no arranque para a criação de um sistema informático.

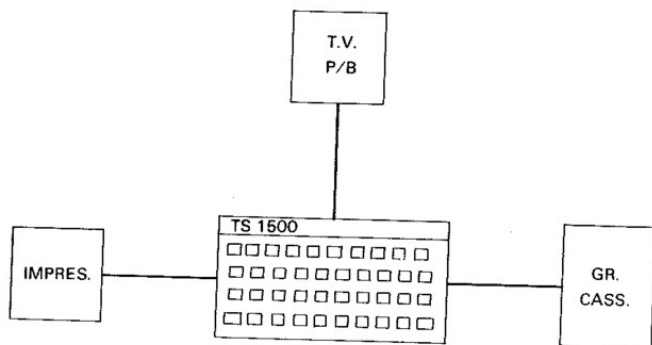


Fig. 7

Este deverá ser constituído pelo TS 1500, um televisor a preto e branco e um gravador. Tratando-se de um computador muito simples e de fácil manejo e programação, deverá ser utilizado para iniciação daqueles que pretendam entrar no mundo da informática (no caso de uma escola, para alunos e professores que pela primeira vez contactem um sistema informático), embora outros computadores sirvam para o mesmo efeito.

Um conselho, apenas: se a Escola decidir adquirir mais unidades, não compre TS 1500, uma vez que todo o desenvolvimento possível (em qualquer âmbito) de uma secção de informática fica definitivamente comprometido pelas sérias limitações desse computador.

O ZX Spectrum

Muitas escolas (sobretudo as não bafejadas com a «sorte» da oferta do TS 1500) decidiram adquirir um (ou mais) Spectrum, com verbas próprias. Dos resultados, nalguns casos espectaculares, de tais decisões, falei em outro capítulo.

Tratou-se, esta sim, de uma decisão inteligente e com largas perspectivas futuras. Porque o ZX Spectrum — embora tendo sido posto no mercado mundial em 1982 — é um computador que mal começou ainda a ser explorado em todas as suas potencialidades. Foi, na verdade, concebido com recurso a tecnologia de ponta e a partir de um projecto tão aberto que todos os meses, desde então, surgem no mercado novos periféricos e programas para o Spectrum. Para se avaliar das suas potencialidades de mercado, bastará dizer que só no ano de 1984 encerraram cinco fábricas de computadores (uma delas nos Estados Unidos), em resultado da impossibilidade de concorrer com o Spectrum nas vendas.

Se é, pois, este o computador ou computadores que a escola possui, é possível com o mesmo montar um sistema como o da figura 8, ou qualquer outro intermédio.

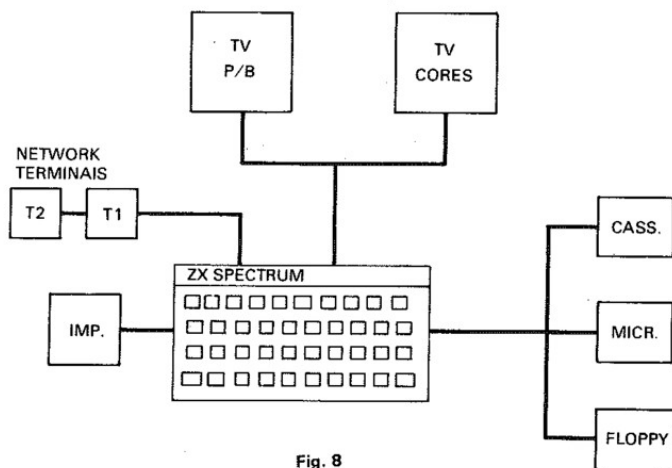


Fig. 8

De qualquer modo (e embora me refira mais desenvolvidamente ao assunto em capítulo posterior), é possível montar um sistema constituído pelo Spectrum, um TV a cores, um gravador de cassetes ou conjunto de microdrives e uma impressora. Não é necessário (e seria até um exagero) adquirir uma impressora para cada um dos computadores que a escola eventualmente possua.

O TC 2068

É provável que, à data de saída deste manual, algumas escolas tenham adquirido o TC 2068 — o «grande» Spectrum da Timex.

Trata-se, na verdade, da versão portuguesa do TS 2068 — computador desenvolvido, em 1983, pela Timex dos E.U.A., baseado na tecnologia do Spectrum e com autorização da Sinclair Research Corporation. Não cabe neste manual uma descrição detalhada desse computador. Todavia, e relativamente ao Spectrum, tem a possibilidade de utilizar 32, 64 ou mesmo 80 colunas de caracteres no écran; três canais de som (além do BEEP), o que permite compor acordes; possibilidade de utilizar as oito cores num só carácter; possibilidade de utilizar écran duplo; e, mais que

tudo isto, um «porto» para inserção de cartridges, o que liberta os cerca de 32K de memória para DADOS.

Na figura 9 pode ser visto um sistema possível a partir do TC 2068.

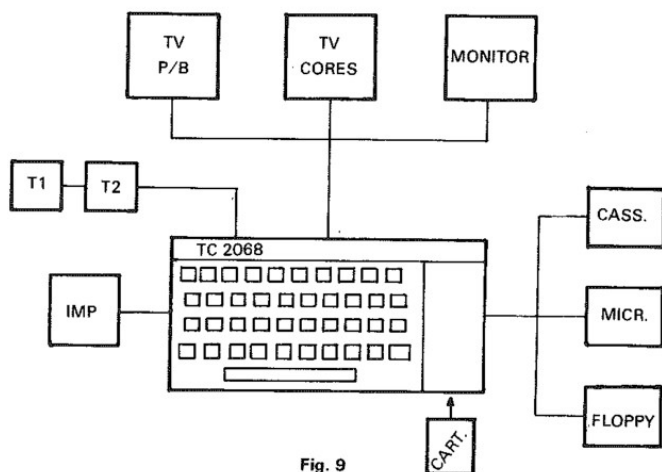


Fig. 9

Este computador é inteiramente compatível com o Spectrum, graças a um emulador que o acompanha e se destina a ser inserido no «porto» para as cartridges.

1.4 — O sistema possível/aconselhável

A decisão (ou imposição) de informatizar uma escola tem implicações muito mais profundas e diversificadas do que as que seriam aplicáveis a qualquer outro tipo de departamento ou empresa.

A escola é uma instituição assente na pedagogia, indispensavelmente virada para o futuro (pelo menos em teoria) e onde, portanto, tudo tem de ser encarado como algo muito diferente de um fim em si mesmo. Queira-se ou não, a escola é apenas o primeiro movimento da roda que inicia um processo acelerado e infinito. Processo esse que tem como sujeitos os alunos: é para eles e por eles que a escola existe.

Assim sendo, seja o que for que se tente implementar em tal instituição, a primeira coisa a fazer é precisamente ter em conta esse sujeito colectivo.

Ora o computador pertence inteiramente à geração dos alunos — razão que leva a que muitos professores se encontrem totalmente desli-

gados do movimento informático e até — o que é mais grave — tentem, de um ou outro modo e por razões, algumas vezes, de insegurança, travar o processo, pelo menos no que diz respeito à entrada de tal inovação na escola. É, contudo, um remar contra a maré — não de um modismo ou contestação juvenil, mas sim da história (coisa que se torna bem mais difícil, e mesmo impossível).

Teremos de considerar, portanto, a entrada definitiva do computador na escola como um dado adquirido e o que me parece mais inteligente é procurar encarar-lo pelo(s) seu(s) lado(s) positivo(s), procurando controlar — não travar — o processo de utilização do mesmo.

Um número considerável de alunos (e, também, de professores) adquiriu e tem em casa computadores ZX Spectrum, trabalhando neles, quer em utilização de programas adquiridos, quer na elaboração de programas novos. Assim, e ainda que outra razão não existisse, parece ser este o computador a adquirir como base de um sistema informático de uma escola. Até porque não impede a expansão do sistema, mesmo com aquisição futura de um ou mais TC 2068.

De resto, é o computador que maior número de programas tem disponíveis no mercado (mais de 20 000, em 1984!), sendo centenas desses programas destinados precisamente ao ensino e alguns deles com qualidade pedagógico-científica notável.

Dispensando-me, embora, de uma descrição exaustiva do ZX Spectrum — em primeiro lugar porque seria inútil para as escolas que já possuem um, e em segundo porque há obras específicas sobre o assunto para os estabelecimentos de ensino queensem adquiri-lo e pretendam obter informação detalhada — apontarei, ainda assim, muito brevemente, algumas das razões que me levam a considerá-lo como o computador ideal para o ensino secundário e preparatório.

Como já atrás disse, trata-se de uma máquina com potencialidades imensas (muitas das quais ainda desconhecidas) e de formato e peso bastante reduzidos, o que torna fácil a sua mobilidade. O preço (cerca de 26.000\$00 na versão de 48K (única que deve ser adquirida, pois a de 16K é muito limitada), aliado às características, torna-o no computador mais barato do mercado. Possuindo uma linguagem extremamente simples (mas poderosa), permite realizar programação com muita rapidez e facilidade — factor decisivo, penso, para a sua aplicação ao ensino. Possui 8 cores de base, além de um comando (BRIGHT) que permite torná-las mais claras ou escuras (na prática obtém-se, portanto, 16 cores). O teclado é extremamente macio, uma vez que é constituído por teclas de borracha. Pode utilizar, como extensores de memória, o gravador de cassettes, a microdrive e as diskettes. Pode ser ligado a qualquer tipo de impressora; plotters; modems, etc. Tem já incorporados gráficos de diverso tipo, podendo ainda ser definidos outros pelo utilizador. O écran apresenta um rectângulo com 32 por 24 linhas, o que dá um total de 768 caracteres.

Para se aquilatar das potencialidades do computador — e, portanto, das suas «limitações» — bastará dizer que, com periféricos já disponíveis no mercado (e que poderão ser progressivamente adquiridos), é possível:

— optando pelo sistema de microdrives, torná-lo extensivo até um número máximo de 8, o que permite acesso rápido a cerca de 720K;

— optando pelo sistema de diskettes da Timex, torná-lo extensivo até ao número máximo de 4, o que poderá vir a permitir acesso directo e aleatório a cerca de 2.000.000 de bytes (2 Mb);

— adquirindo a Interface 1 (transformador), montar uma rede de Spectrums interligados, até ao número de 64.

Todas as demais hipóteses postas em 1.2 são possíveis com o Spectrum.

Para cada ZX Spectrum deverá ser adquirido um televisor⁽¹⁾ e um gravador de cassettes dos mais baratos⁽²⁾; e, para o conjunto, pelo menos uma impressora.

Quanto ao TC 2068, sem dúvida que é uma meta a atingir pela escola, quando e se tiver disponibilidades para tanto. Dispensamo-nos de indicar as suas vantagens, mesmo sobre o Spectrum, uma vez que já anteriormente foquei as mais importantes. Para além de que, repito, é inteiramente compatível com o ZX Spectrum, razão pela qual permite aproveitar e desenvolver todo o trabalho executado neste último.

⁽¹⁾ É evidente que as potencialidades do Spectrum só serão integralmente aproveitadas com televisores a cores. Não é menos evidente, contudo, que o preço dos TVs a cores torna impensável — atendendo à realidade económica das escolas do nosso país — a aquisição de um TV a cores para cada computador. Contudo, é aconselhável adquirir pelo menos um, quanto mais não seja para utilização colectiva, em determinadas actividades. Para os restantes, podem ser adquiridos TVs a preto e branco.

⁽²⁾ De igual modo, não é provável a existência de capacidade económica que permita a aquisição de um sistema de microdrives (e muito menos de diskettes) para cada Spectrum. Contudo, é igualmente aconselhável a aquisição de, pelo menos, um conjunto para programas que necessitem de tratar com rapidez elevada quantidade de informação.

COMO MONTAR UMA SECÇÃO DE INFORMÁTICA

- 2.1 — A sala
- 2.2 — O equipamento
- 2.3 — A instalação
- 2.4 — Alguns cuidados básicos
- 2.5 — A falta de verba...

2.1 — A sala

Atendendo à superlotação vulgar nas nossas escolas, não valerá a pena falar em salas ideais (como, de resto, em nada ideal, já que esta é uma meta sempre inalcançável). Todavia, e apesar de normalmente se «escolher» a sala disponível (frequentemente uma que não sirva para aulas normais ou outras actividades consideradas prioritárias), importa referir algumas das *condições mínimas* que uma sala destinada à informática deverá possuir.

Assim, deverá pelo menos ter-se o cuidado de escolher uma sala sem humidade (ou com um índice muito baixo), se se querem evitar problemas futuros, não apenas de eventuais falhas no funcionamento do equipamento, como, e sobretudo, na sua deterioração. Não há maior inimigo do material electrónico do que a humidade, nomeadamente de fitas magnéticas — armazéns de todo o trabalho realizado.

O problema da temperatura não se põe para sistemas tão pequenos. Se esse é um problema dos grandes computadores, que, normalmente, são instalados em salas climatizadas, o índice de aquecimento de uma ou duas dezenas de Spectrums não constitui problema relevante.

A dimensão da sala dependerá da opção que se tiver feito, sobretudo quanto ao número de unidades que o sistema contenha, ou venha a conter (é sempre conveniente ter uma visão da expansão futura do mesmo). Deve, contudo, ter-se em atenção a conveniência de existir um espaço razoável entre cada mesa, por forma a evitar encontrões que em nada beneficiam o trabalho e funcionamento dos sistemas.

Ao contrário do que é desejável para uma aula normal, a iluminação da sala não deverá ser excessiva, não constituindo mesmo qualquer problema o facto eventual de ser insuficiente para trabalho normal de

outro tipo. De qualquer modo, deve evitar-se a existência de luz forte e incidente nos ecrãs.

Quanto à segurança... isso dependerá das possibilidades e opções que cada escola, face às disponibilidades existentes, decida tomar.

2.2 — O equipamento

A) Da sala

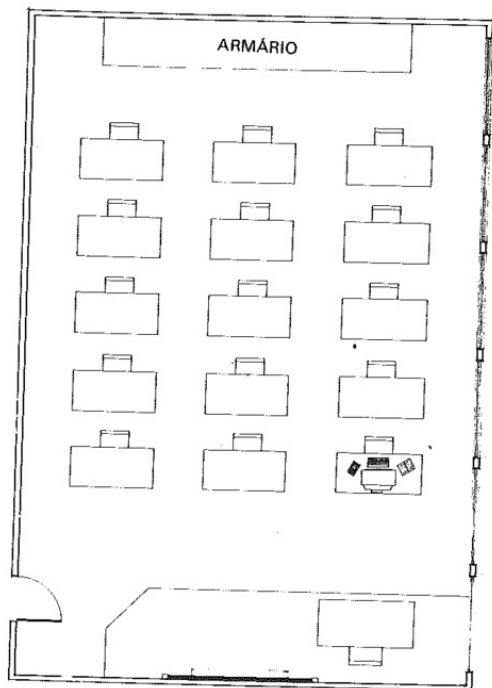


Fig. 10

O equipamento necessário para a sala (Fig. 10) é muito simples: uma mesa (não uma carteira inclinada!) para cada conjunto, suficientemente espaçosa para comportar o computador, o televisor, o gravador, e ainda livros e blocos; e um armário (ou vários, dependendo do tipo de arrumação) destinado a guardar os sistemas, as cassettes e os livros, além de outro material julgado necessário.

B) Dos sistemas

B.1 — Hardware (parte física do sistema)

Dependendo, claro, da opção tomada, será constituído por um certo número de computadores, igual número de televisores (ou monitores), gravadores de cassette (eventualmente microdrives ou diskettes, para as escolas que tenham essa possibilidade) e uma impressora. Deverá ainda ser adquirido um certo número de cassettes virgens (C 12 são, quanto a mim, as mais aconselháveis, por permitirem guardar um programa em cada lado de cada uma, o que torna mais fácil organizar o arquivo e o acesso independente a cada programa), no mínimo três para cada conjunto.

B.2 — Software (programas)

Neste campo não há regras fixas. O computador vem acompanhado com uma cassette de demonstração (muito útil, de resto, no lado A, uma vez que é profundamente explicativa sobre todo o sistema Spectrum; no lado B contém não só jogos como alguns programas utilitários e pedagógicos). Para além disso, a escola irá adquirindo os programas que entender, entre os muitos disponíveis no mercado, como referirei em pormenor na Parte II.

B.3 — Manuais e livros

É importante que, para além dos manuais que acompanham o computador, a escola vá adquirindo livros (e revistas) especializados e que constituem um apoio *imprescindível* ao desenvolvimento do trabalho, permitindo uma actualização permanente dos utilizadores. Em anexo indicarei os títulos de algumas obras aconselháveis.

2.3 — A instalação

A) A parte eléctrica

Teoricamente (e na prática, mesmo) é possível ligar o sistema (computador, televisor e gravador) a uma mesma tomada, com uma ficha tripla. E é de resto o que a maior parte das pessoas faz, em suas casas, quando monta um sistema informático.

Contudo, verifica-se, por vezes, uma certa inconveniência em efectuar este tipo de ligação, pois acontece surgirem interferências (nomeadamente variações de tensão bruscas) que apagam parcialmente (ou alteram) as memórias RAM do computador.

Assim, a forma correcta de preparar a instalação eléctrica é fazê-lo por forma a fornecer dois circuitos independentes a cada posto de trabalho (Fig. 11). A um dos circuitos deve ligar-se *apenas o computador*; ao outro podem ligar-se, sem qualquer problema, todos os demais periféricos.

Sendo o consumo do sistema diminuto, não é necessário instalar

qualquer contador ou disjuntor especial, podendo os circuitos referidos partir de duas caixas de derivação vulgares.

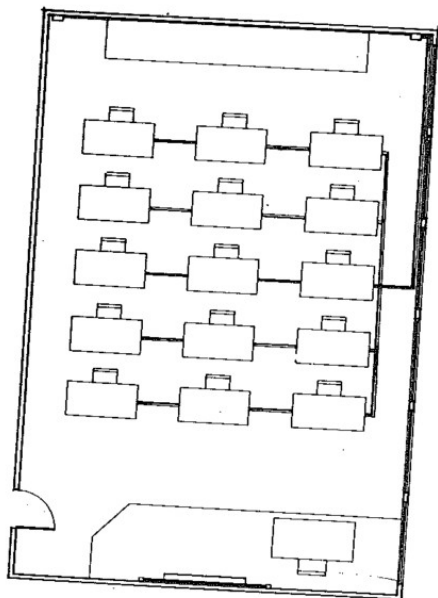


Fig. 11

B) Os postos

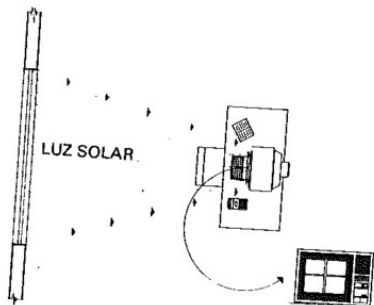


Fig. 12

A instalação dos postos de trabalho, se bem que simples, aconselha alguns cuidados relacionados essencialmente com o utilizador.

A regra de ouro é, como pode ver-se pela figura 12, não permitir que incida qualquer fonte de luz directamente sobre o écran, a fim de evitar reflexões que dificultariam a captação de informação, para além de cansar a vista.

A solução também não é, por razões igualmente visíveis na figura 13, a de colocar o sistema em sentido inverso ao das janelas.

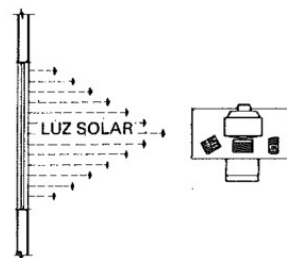


Fig. 13

Tal como elucidada a figura 14, a instalação correcta é aquela em que não haja janelas à frente nem atrás dos postos, mas sim do lado esquerdo.

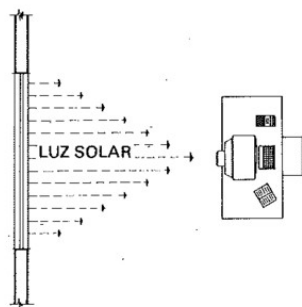


Fig. 14

Esta colocação, além de impedir a incidência de raios luminosos no écran ou na vista do utilizador, permite iluminação suficiente para a consulta de livros ou manuais (à esquerda do operador), bem como para a tomada de notas num bloco.

Em cima da mesa (não carteira inclinada, repito), o material ficará disposto como o indicado na figura 14: o televisor ao centro e na parte mais afastada; o computador nesse alinhamento e junto do utilizador; o gravador à direita, bem como as cassettes que estão a ser utilizadas; livros, revistas, manuais e blocos à esquerda (podendo colocar-se o bloco à direita, o que permite maior facilidade na escrita).

C) O armário

É evidente que a maioria das escolas não irá, por certo, mandar construir armários específicos para a arrumação dos sistemas. Certamente aproveitarão armários disponíveis, procurando utilizá-los da melhor forma.

Contudo, na figura 15, apresenta-se um tipo lógico possível de armário (ao menos para as escolas que, de facto, pretendam mandar fabricá-lo ou como ideia de arrumação para as outras).

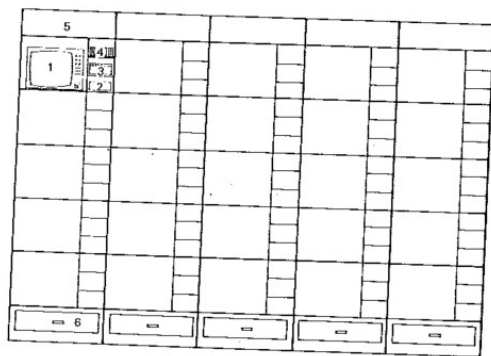


Fig. 15

- 1 — MONITOR
- 2 — COMPUTADOR
- 3 — GRAVADOR
- 4 — CASSETTES OU OUTROS SUPORTES
- 5 — LIVROS, DOCUMENTAÇÃO E SOFTWARE
- 6 — CABOS, FICHAS, ETC.

2.4 — Alguns cuidados básicos

Embora o sistema não seja constituído por material especialmente delicado, há sempre algumas verdades que, mesmo sendo, eventualmente, de «La Palisse», se ditas, evitarão por certo alguns dissabores ou até consequências mais graves.

Já falámos da humidade que, a qualquer custo, é necessário evitar, relativamente a qualquer dos componentes do sistema.

O computador é uma máquina sem elementos móveis, pelo que não põe qualquer problema especial, nem mesmo de transporte. Pode ser manipulado à vontade, sem receio de que algo «lá dentro» se quebre ou solte.

Com o televisor já se não passa o mesmo. Jamais deve transportar-se um televisor imediatamente após ter sido desligado. Se bem que o seu interior seja constituído por elementos sólidos (como o computador), possui uma enorme válvula (o cinescópio) com filamentos que aquecem ao rubro e podem, com qualquer movimento brusco, deteriorar-se. Assim, convém deixá-lo arrefecer um pouco antes de o arrumar no armário, para além de um certo cuidado ao pousá-lo.

Com as cassettes (ou microdrives, ou diskettes) é necessário um cuidado suplementar. Trata-se de suportes magnéticos que, como tal, podem alterar-se com a proximidade de campos magnéticos ou eléctricos com indução. Assim, *nunca devem ser colocadas* junto do transformador de alimentação nem do aparelho de televisão (sobretudo na parte de trás), uma vez que este gera um campo electromagnético fortíssimo quando ligado. Podem, contudo, perfeitamente ser colocadas sobre a mesa e a uma distância não inferior a 30 centímetros de qualquer das fontes referidas. Não é provável que alguma das cassettes fique muitos meses sem ser utilizada. Se tal acontecer, contudo, é conveniente ao menos rodá-la em alta velocidade para um lado e outro no gravador, para evitar alterações da informação magnética que contenha, devido à posição de «acamado» da fita.

Resta falar da limpeza do material.

Todo o equipamento electrónico é atreito a acumular rapidamente uma quantidade considerável de poeira, uma vez que esta sofre uma forte atracção dos campos electromagnéticos gerados pelo funcionamento de tais equipamentos. Tal facto nota-se especialmente nas televisões, pois trata-se de máquinas que chegam a produzir correntes da ordem dos 200 000 volts — o que cria um campo magnético fortíssimo.

É hábito das pessoas limparem o pó de móveis, livros e utensílios com panos, hábito esse generalizado nas escolas — onde, ainda por cima, a quantidade de material a limpar, aliada ao número reduzido de pessoal e às limitações de tempo, não torna esse tipo de trabalho delicado.



Fig. 16

Na figura 16 podem ver-se alguns dos inconvenientes da utilização de panos para limpar o equipamento electrónico: as teclas dos computadores sofrem torções violentas, que muito rapidamente as danificam; os ecrãs dos televisores são riscados diariamente e, embora de forma invisível (inicialmente), acabam com sérios danos, que se reflectem sobretudo na vista dos utilizadores; etc.

E o pior de tudo é que, tendo o equipamento em causa ranhuras, vincos e interstícios, o pano não consegue limpar tais zonas — algumas das quais são portas de entrada da poeira para o interior dos circuitos.

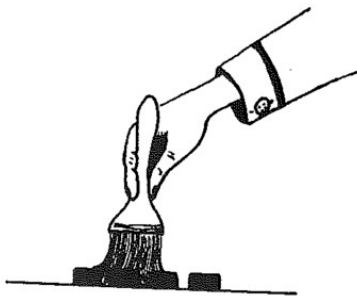


Fig. 17

Assim, e como pode ver-se na fig. 17, o utensílio ideal para a limpeza do equipamento é um simples pincel de cordas finas, com o qual se evitam todos os inconvenientes apontados, garantindo-se, por outro lado, uma limpeza perfeita.

O equipamento deverá, de preferência, ser escovado diariamente — no fim de cada sessão de utilização, podendo essa tarefa ficar a cargo do utilizador.

2.5 — A falta de verba...

Até aqui, e por diversas vezes, tenho vindo com alguma frequência a focar o problema da falta de verba, como condicionante ao desenvolvimento, mais ou menos rápido, de uma secção de informática. E se o faço é tão só por constituir uma realidade (difícilmente ultrapassável) para as escolas, cujas verbas disponíveis nem sequer chegam, muitas vezes, para o indispensável. Todavia, é precisamente nas dificuldades que surgem, muitas vezes, as soluções; nomeadamente quando há um desejo e uma vontade grandes para levar por diante um projecto. E se alguns acabam por mostrar-se inviáveis — atendendo até à falta de adesão por parte do corpo da escola — outros realizam-se com uma facilidade surpreendente. É o caso, segundo penso, da montagem de uma secção de informática.

E para que se não julgue que isto não passa de teoria, vou descrever um exemplo de uma escola que o fez e, neste momento, o mantém em sólida expansão.

No ano lectivo de 1983/84, professores e alunos de matemática da Escola Secundária de S. Julião, em Setúbal, efectuaram uma visita à Faculdade de Ciências e Tecnologia, em Almada, a qual possui uma dinâmica secção de informática. Dessa visita resultou a ideia inicial de utilizar os microcomputadores na disciplina de matemática, no ensino secundário — mais propriamente, na E.S.S.J.

De imediato se constituiu um núcleo — com professores e alunos — subordinado à ideia «Computadores em S. Julião, porque não?», núcleo esse que desenvolveu um projecto de aquisição e instalação do equipamento necessário à realização do mesmo.

Tendo-se decidido — após análise e estudo de vantagens e inconvenientes (sob vários aspectos) dos diversos equipamentos disponíveis no mercado — pela adesão a um sistema baseado no ZX Spectrum, projectaram, desenharam e mandaram imprimir 2000 autocolantes destinados à recolha de fundos necessários para a aquisição das primeiras unidades. Em fins do ano lectivo de 1983/84 realizaram uma rifa, com idêntico objectivo.

Por essa altura, e paralelamente, professores delegados ao Conselho Pedagógico da Escola fizeram uma proposta para a constituição de uma *Secção de Vídeo e Informática*, proposta que foi aceite e da qual resultou a compra de 1 TV a cores, 1 TV a preto e branco, 1 equipamento vídeo completo (com câmara) e 2 computadores ZX Spectrum — material este adquirido no ano lectivo seguinte.

Numa primeira fase formou-se, pois, uma secção mista — vídeo e computadores — sendo a mesma constituída por professores dos diversos grupos. Posteriormente, a secção subdividiu-se: uma subsecção de vídeo e uma subsecção de computadores. Esta passou a contar também com alunos, no seu núcleo, e a dispor de verbas próprias — independentes do orçamento da escola —, que gere autonomamente.

Para além da aquisição de material e recolha de mais fundos, a actual secção de computadores da escola programou e realizou já um curso de BASIC (ao qual me referirei noutro capítulo). Neste momento procura obter uma sala para instalação do equipamento em moldes definitivos, a qual servirá essencialmente para os tempos livres. Prioritariamente destinada à aprendizagem do BASIC, bem como ao trabalho de investigação e programação informática, a sala irá servir, também, eventualmente, para fins lúdicos.

No fim do ano lectivo de 84/85, a secção dispunha já de uma verba com a qual pensava adquirir uma impressora e mais uma unidade de computador.

UM PROJECTO INTERESCOLAS

- 3.1 — Uma rede entre escolas
- 3.2 — Um banco de dados comum
- 3.3 — O futuro

Nenhum livro — nem mesmo uma gramática que, em princípio, tem como função essencial sistematizar o «status» de um código linguístico — deve limitar-se à sistematização do que já existe. Ao contrário, penso que qualquer obra — mesmo (ou sobretudo) pedagógica — deve ser produzida com um pé no futuro, não especulativo, claro, mas viável. Caso contrário torna-se, no momento mesmo da publicação, mais uma obra de biblioteca, a ser consultada raramente e com necessidades mais ou menos históricas.

Esta a razão que me leva a incluir este capítulo. Projecto arrojado — dirão alguns —, mas inegavelmente possível e, como se verá, de grande utilidade. E se em certas zonas do país tal projecto pode parecer ficção, outras há em que é possível pô-lo de imediato em prática, com todos os benefícios inerentes.

3.1 — Uma rede entre escolas

Mesmo com as limitações económicas — que nunca é demais sublinhar — da maioria dos estabelecimentos de ensino, é possível criar uma rede informática entre as escolas de, por exemplo, uma cidade.

Pondo de parte, por agora, um sistema mais evoluído que adiante referirei, não há qualquer inconveniente em estabelecer uma ligação coordenada entre as secções de informática das escolas de uma localidade. Pelo contrário, tal ligação pode tornar-se altamente vantajosa para todas elas.

Em primeiro lugar, permitiria a aquisição diversificada de software, o qual poderia ser então permutado à medida das necessidades de cada escola. O mesmo se passaria com obras sobre informática: sendo conveniente que algumas delas fossem adquiridas por todos os estabelecimentos, outras há que poderiam ser de uso comum.

Nada impediria, igualmente, a utilização do próprio equipamento, de forma planificada, por alunos ou professores das diversas escolas, de

modo a evitar o seu subaproveitamento, nuns casos, e insuficiências, noutros.

Mais importante, porém, seria a permuta de informação referente a trabalho de investigação e desenvolvimento de programas e rotinas, bem como encontros inerentes ao trabalho desenvolvido em cada uma das escolas. Para além de troca de experiências, tais encontros permitiriam nivelar conhecimentos de forma mais fácil e prática, dotando cada escola com uma riqueza sempre actualizada de informação e permitindo, em última análise, projectar e realizar trabalhos comuns, por certo de muito maior dimensão e com resultados de âmbito mais vasto e aprofundado.

Todo este processo, aliado a uma possível melhoria económica, virá permitir a criação de uma rede de comunicação via linha telefónica, em fase mais adiantada.

Para tanto bastará que cada escola adquira um «modem» (periférico precisamente destinado ao estabelecimento de contactos telefónicos entre computadores). A partir daí estão criadas as condições para a troca de informação, dados e programas, sem necessidade de deslocações ou outros suportes. Cada escola poderá enviar directamente para as memórias (ou periféricos) dos computadores das outras escolas toda a informação que desejar, bem como recebê-la, através das linhas telefónicas e instantaneamente.

E é até possível, por exemplo, programar determinadas sessões em conjunto, emitindo-as de uma para todas as escolas da cidade, ao mesmo tempo; com possibilidades de intervenção de qualquer das escolas, em qualquer momento, permitindo assim o diálogo.

3.2 — Um banco de dados comum

Criada uma rede do tipo da anteriormente descrita, com um pequeno passo mais será possível desenvolver um banco de dados comum.

O que é um banco de dados? «É um conjunto de informações directamente exploráveis, geralmente estruturadas em base de dados e abrangendo um domínio particular do conhecimento.» ⁽¹⁾

Será possível, por exemplo, desenvolver um banco de dados de Biologia, ou Linguística, ou Desenho, etc., contendo um conjunto de informação programada pelos professores das diversas escolas e que permita o acesso a alunos ou professores, em determinados momentos do dia. De resto, correctamente planificado, esse acesso poderá ser distribuído pelas várias disciplinas, em diversas horas e durante lapsos de tempo determinados.

Para a criação de um tal banco com um nível razoável de informação, torna-se indispensável a aquisição de um conjunto de Floppy Disk. O FDD da Timex Portugal é acessível, pode conter até quatro drives (quatro discos em simultâneo), brevemente de dupla face e 80 pistas, o que, na sua máxima combinação, permite o acesso muito rápido a cerca de 2 milhões de bytes de informação — número muito razoável para a criação de um banco de dados como o que acima se descreve.

⁽¹⁾ Dictionnaire de l'informatique

Tal sistema poderá ser adquirido pelas escolas, em conjunto, ficando instalado em uma delas (ou, até, de forma rotativa, em todas) e sendo acedido via «modem» por qualquer das outras.

As vantagens e possibilidades que tal banco proporciona são por demais evidentes.

3.3 — O futuro

À semelhança do que já sucede em outros países (Inglaterra, França, Brasil, Espanha, etc.), não serão muitos os anos passados até que se criem bancos de dados nacionais, com capacidade de informação da ordem dos milhões de Megabytes, permanentemente actualizados e acessíveis a todos os assinantes de tais sistemas.

Habitualmente são empresas privadas quem se encarrega da montagem e organização desses bancos, se bem que em alguns países (França, por exemplo) alguns departamentos estatais mantenham bancos de dados destinados aos serviços públicos. Mediante o pagamento de uma assinatura mensal (ou anual), cada assinante passa a ter acesso a esses gigantescos bancos e a toda a informação que os mesmos contém.

A par de bancos de dados de interesse comercial e industrial, têm-se desenvolvido fundamentalmente outros de âmbito científico e pedagógico, para escolas e universidades, abrangendo as mais diversas ciências e disciplinas. Por um preço muito menor do que o que custariam os livros necessários para obter, de forma actualizada, um tal volume de informação, é possível, pois, de forma simples e rápida, manter em dia não apenas os conhecimentos nos vários sectores como ainda estar a par, permanentemente, das últimas descobertas nos diversos campos do conhecimento humano.

Resta dizer que essas firmas — pelo menos as de maiores dimensões — estabelecem redes mundiais de bancos de dados, o que permite o acesso a qualquer tipo de informação em qualquer momento e em qualquer ponto do globo.

II PARTE

A UTILIZAÇÃO DO SISTEMA

QUE FAZER COM O SISTEMA?

- 1.1 — Um novo método de trabalho
- 1.2 — Tempos e ocupações da sala
- 1.3 — Como começar

1.1 — Um novo método de trabalho

Um dos habituais receios que acompanha a entrada em funcionamento de um sistema informático em qualquer local de trabalho é a possibilidade de o mesmo pôr em causa métodos até aí seguidos e, mais do que isso, interiorizados, ou até, em última instância, alguns postos de trabalho. Receio infundado, mas legítimo, apesar de tudo.

Todavia, sendo o computador uma extensão do ser humano, criada e aperfeiçoada por este, é óbvio destinar-se fundamentalmente a permitir-lhe aumentar as suas capacidades de trabalho — sobretudo o *criativo* — encarregando-se *exclusivamente* de cumprir as suas ordens (quer dadas directamente, quer em forma de programas). E, o que não é menos importante, desempenhando a alta velocidade e sem qualquer erro as tarefas repetitivas que ao homem provocam rapidamente cansaço.

Ou seja: em vez de o ser humano ter de ir executando qualquer trabalho, à medida que calcula resultados — obrigado, após conhecimento dos mesmos, a re-planificar e corrigir (o que atrasa, entre outras coisas, a conclusão desse trabalho) —, pode desenvolver antecipadamente um projecto de execução desse mesmo trabalho, planificado e testado até à perfeição, transmitindo o plano de instruções daí resultantes de uma só vez ao computador, que sem falhas nem paragens, o executará ao ritmo e com a profundidade indicados.

Digamos, portanto, que o ser humano fica liberto para tarefas de análise, síntese e planificação — tudo isto ao nível do raciocínio e criatividade, enquanto o computador se encarrega da simples execução. Isto é, o ser humano fica liberto — em tempo e qualidade de trabalho — da inutilidade (imprescindível, é certo) do simples cálculo e execução. Numa comparação ousada, diria que cada operador de computador se transforma quase num gestor de determinada tarefa, com um tranquilo e incansável auxiliar às ordens: o computador.

De tudo isto resulta que a entrada do computador em qualquer local de trabalho obriga, de facto, à alteração de métodos, mas em sentido positivo, tanto no que respeita à produtividade como — e o que é mais importante — à qualidade do trabalho.

Vejamos o que habitualmente acontece num escritório, por exemplo, antes da entrada em funcionamento de um sistema informático.

É claro que existe uma planificação do trabalho, bem como uma distribuição deste por cada empregado. Assim, há um funcionário que se encarrega de lançar documentos (facturas, recibos, etc.), os quais terão que ser registados em folhas próprias para o efeito. Para efectuar esses lançamentos, é necessário consultar tabelas (de preços, por exemplo), efectuar cálculos (dependendo de quantidades, descontos, etc.), conferir e arquivar. Além disso, é necessário ir actualizando stocks à medida que determinado artigo entra ou sai; actualizar preços (por vezes todas as semanas); fazer notas de encomenda; enfim, um sem número de tarefas.

É, certamente, anti-económico manter dois ou três empregados na realização desse trabalho, quando um pode executá-lo. Mas se esse um fica doente, por exemplo, grandes dificuldades se deparam, de ordinário, precisamente por não existir outro com o mesmo grau de conhecimentos acerca dessas tarefas específicas. Isto para não falar nas férias.

Que alteração se verifica, após a informatização? O mesmo funcionário (agora operando o computador) faz *apenas um* lançamento. É o novo «colega» que se encarrega, com uma rapidez e exactidão notáveis, de actualizar stocks, emitir avisos de mínimos, imprimir o documento, fazer todos os cálculos, etc. O fulcro do trabalho humano deslocou-se assim para a *programação* das tarefas a executar, bem como para a preparação dos documentos a serem tratados pelo computador.

Antes de mais, portanto, a utilização do computador permite uma planificação profundamente pensada e testada das tarefas a executar; por outro lado, abre possibilidades a uma maior individualização do trabalho — o que resulta, inevitavelmente, em menor percentagem de falhas e em maior qualidade.

No que respeita especificamente ao ensino, sem dúvida que a função do professor sofre uma profunda alteração. Todavia, não como muitos pensam e afirmam: — «qualquer dia não são necessários professores, pois o computador substitui-os...». Bem pelo contrário, o papel do professor torna-se ainda mais importante; só que, se por um lado o liberta para uma programação mais correcta e criativa das actividades lectivas, por outro o obriga a uma actualização mais permanente. E, o que é mais importante, liberta-o — em termos temporais e não só — para um mais perfeito método de avaliação do aluno.

De facto, de agente de ensino, que de algum modo é neste momento, o professor passa, a par do aluno, à condição de sujeito — como noutro capítulo poderemos ver.

O aluno, por seu lado, entre outras coisas, passa a possuir um auto-controlo muito maior e mais aperfeiçoado sobre a sua progressão na aprendizagem, tornando-se igualmente agente activo do processo.

Estas alterações (e outras) na metodologia do trabalho, resultam também inevitavelmente em alterações muito positivas no relacionamento

professor/aluno — com tudo o que daí decorre, enfim, para o funcionamento das aulas e do próprio estabelecimento de ensino.

E, para que tudo isto resulte, basta que ambos — professor e aluno(s) — ponham ao serviço do projecto a sua criatividade, a sua capacidade de diálogo (que, de qualquer modo, fica automaticamente amplificada) e o entusiasmo que certamente qualquer deles possuirá.

1.2 — Tempos e ocupações da sala

Vamos admitir que a escola considera realmente importante a criação de uma secção de informática, reconhecendo as vantagens, de diversa ordem, que o desenvolvimento de tal secção tem, na realidade. E, consequentemente, que faz mesmo um esforço para reservar uma sala para tal secção — apesar da superlotação e dificuldades de horários das restantes sessões lectivas.

(É bem possível que, em muitas escolas, inicialmente se não veja como ocupar uma sala, na ocasião de abertura do estabelecimento de ensino. Todavia, muito depressa se surpreenderão com o índice de ocupação do tempo.)

Desde logo é necessário planificar a utilização da sala, não apenas em termos de horários como também do tipo de actividades que ali se desenrolarão. Trabalho que não é difícil nem polémico, já que pelo que tenho verificado, há sempre uma meia dúzia de professores (e muito mais que meia dúzia de alunos) entusiasticamente dispostos a colaborar em tal actividade.

Não é de crer que, pelo menos na maioria das escolas, neste momento haja já muitos professores preparados — e predispostos — para utilizar tal sala em aulas «normais». Assim sendo (e ao contrário do que por vezes sucede, por exemplo, com a sala de audiovisuais), importa seriar, como dissemos, o tipo de ocupação.

Para uma escola que decidiu recentemente adquirir o sistema informático, dois tipos de ocupação se impõem desde logo:

1 — Programar e ministrar, a alunos e professores, um curso básico, muito simples e breve, sobre informática.

2 — Prever algumas horas por semana para utilização, por alunos e professores, com o objectivo de desenvolver programas, efectuar experiências e iniciar algumas investigações nos domínios da informática.

Sendo embora possível, apesar de tudo, que haja um ou outro professor a pretender utilizar de imediato a sala para a realização de uma ou outra aula, as demais actividades poderão ir entrando progressivamente nos tempos disponíveis. Uma, das talvez não menos importantes — os tempos livres —, é de garantido sucesso (a vários níveis) e contribuirá, decisivamente, para uma mais rápida e segura evolução da secção de informática da escola. A esta actividade me referirei no capítulo seguinte, exemplificando com duas experiências realizadas no passado ano lectivo, em duas escolas secundárias.

1.3 — Como começar

No início da minha breve — três anos — carreira de jornalista (antes de me dedicar ao ensino), senti grandes dificuldades, na primeira vez que tive de escrever uma grande reportagem. Aliás, o problema de *começar* seja o que for, de vencer a inércia, é sempre um tanto ou quanto pesado, mesmo para o ser humano. Na altura, e após ter começado o parágrafo inicial da reportagem uma quantas vezes; de ter tirado violentamente o papel da máquina outras tantas; e de o amassar com alguma fúria e frustração, o meu chefe de redacção perguntou-me o que se passava.

— É que... não consigo encontrar maneira... enfim, a melhor, para começar isto! — desabafei.

A resposta do Rodrigo Pinto — tal era o nome dele —, simples e rápida, serviu-me para sempre:

— Comece de qualquer maneira, mas comece. Se está à procura da melhor... *nunca* a vai encontrar, *antes*.

E é assim, de facto. O fundamental é vencer a inércia. Neste caso, tirar o(s) computador(es) do cofre, instalá-lo(s) e permitir a sua utilização por quem já saiba. Isto, no que respeita à escola.

Para quem utiliza um microcomputador pela primeira vez, não deixo de apontar algumas orientações — necessariamente breves, já que existem manuais e obras exclusivamente dedicadas ao assunto.

No que diz respeito aos computadores da Timex — TS 1500, ZX Spectrum e TC 2068 — deve começar *sempre* por ler as primeiras páginas dos respectivos manuais, uma vez que as mesmas se referem, com bastante clareza, à forma de ligar o computador, sintonizar o televisor, ligar o gravador, etc. Logo de seguida — e antes de efectuar qualquer experiência de programação aconselho fortemente a introduzir a casete de demonstração, a qual, como verão, apesar de simples, contém uma iniciação muito clara e bem estruturada, que permite a qualquer pessoa entender de imediato com que tipo de equipamento está a lidar, a sua constituição, etc.

A partir daí, pode começar por utilizar o manual do computador ou a obra «Como programar o seu ZX Spectrum»; qualquer deles constitui, nas primeiras 50 páginas, um processo de iniciação bastante simples de começar a verificar, na prática, como trabalha o novo equipamento.

Para além dessas páginas, o manual dá algumas indicações genéricas mas fundamentais sobre algumas das possibilidades básicas do computador, nas suas diversas capacidades, sempre com pequenas rotinas ou programas que permitem ao «aprendiz» ir pondo em prática e verificando a teoria, à medida que vai tomando conhecimento da mesma.

Alguns manuais — como é o caso do manual para o TC 2068 — vão muito além disso, acabando mesmo por constituir um resumido curso de BASIC.

Em apêndices, essas obras possuem um dicionário das palavras-chave, com as funções e atributos de cada uma, bem como breve indicação dos efeitos que produzem — o que torna o manual uma obra de consulta permanente, mesmo para os «iniciados». Possuem também os códigos do conjunto de caracteres utilizados pelo computador, bem como das variáveis do sistema, mapas de memória, etc.

2

ALGUMAS OCUPAÇÕES POSSÍVEIS

2.1 — Os cursos básicos

2.2 — Programação

2.3 — Aulas

2.4 — Tempos livres e clube de informática

Não pretendo, neste capítulo, esgotar todas as possibilidades de utilização dos sistemas informáticos nas escolas — tanto mais quanto é certo existirem mesmo outras utilizações, fora do âmbito pedagógico, que não serão tratadas nesta obra.

De qualquer modo, indicarei, de forma sistematizada, as ocupações básicas dos sistemas, ou seja, aquelas que são inerentes à aquisição dos mesmos, bem como outras que, constituindo possibilidades não obrigatórias, verifiquei serem de grande utilidade, sob diversos aspectos, na vida escolar.

2.1 — Os cursos básicos

Tal como disse no fim do capítulo anterior — e também no início da obra — o computador possui uma linguagem, o BASIC — que utiliza para a sua comunicação com o ser humano. Essa linguagem permite-lhe receber ordens inteligíveis e, em função de cada «vocabulo» (palavra-chave), executar determinada tarefa.

Por outro lado, existe uma sintaxe para essa linguagem — combinações possíveis entre palavras-chave — que foi estabelecida pelos fabricantes.

Tal como para comunicar — em português ou em outra qualquer língua — necessitamos conhecer o vocabulário (palavras-chave) e sintaxe (quer esse conhecimento seja empírico, quer seja estruturado), sob pena de não conseguirmos transmitir nem receber compreensivelmente as mensagens, o mesmo se passa em relação ao computador. Não é possível, pois, estabelecer comunicação sem um conhecimento básico destes dois elementos: *vocabulário e sintaxe*.

Contudo, a qualidade da comunicação e dos consequentes resultados vai depender não apenas do grau de conhecimento dos elementos referidos como ainda da capacidade para «descobrir» e até «inventar» novas possibilidades de combinações sintáticas, entre outras coisas. Um pouco como na obra literária, em que o escritor — conhecedor, como outros falantes da língua, de vocabulário e sintaxe — *cria*, com os mesmos vocabulários disponíveis, não só vocábulos derivados como ainda a utilização sintática de formas não utilizadas na linguagem corrente.

Os manuais — que como atrás disse são úteis e imprescindíveis na iniciação e posteriormente — constituem, contudo, pouco mais que «gramáticas» e «dicionários» pragmáticos do vocabulário e sintaxe do computador. E, assim como se não encontram em qualquer dicionário ou gramática de uma língua os desenvolvimentos literários possíveis da mesma, também os manuais dos computadores não possuem as regras de criatividade (o que, de resto, seria um contra-senso!) para o trabalho com o equipamento a que se referem. Além do que, apesar de simples, não se encontram habitualmente redigidos em linguagem verdadeiramente universal.

Tudo isto leva a concluir que é de grande utilidade a organização, na escola, de cursos, necessariamente breves, de iniciação à informática.

Como atrás disse, há sempre meia dúzia de professores e alunos já possuidores de micros, e frequentemente com graus de conhecimento

mais ou menos elevados acerca dos mesmos. Tais elementos poderão (e, habitualmente, desejam fazê-lo) organizar os referidos cursos, que constituem, na sua base, explicações *práticas* e progressivas sobre «o que são computadores»; «como funcionam»; «o que podem ou não fazer»; etc.

Não há regras fixas para a organização de tais cursos, e a sua planificação deve mesmo ser deixada ao cuidado dos que irão desempenhar as funções de monitores dos mesmos. Todavia, e por algumas experiências práticas de que tenho conhecimento, fica a indicação: o mínimo deveria ser um curso com duas sessões de uma hora cada, semanalmente, durante pelo menos um mês — e, repito, apenas para uma iniciação. Caso haja disponibilidade e maior ambição, poderá organizar-se um curso realmente importante com a duração de seis meses.

De qualquer modo, aqui se deixa um exemplo de curso organizado e realizado numa escola secundária, o que terá sempre o mérito de pelo menos constituir uma base de trabalho para quem não tem qualquer ideia sobre o assunto.

Escola Secundária de S. Julião — Setúbal
Secção de Computadores
Cursos de iniciação ao Basic (Spectrum)

1. *Objectivos:*

- iniciar professores, trabalhadores auxiliares/administrativos e estudantes na programação Basic.
- utilizar os microcomputadores, através da construção de pequenos programas, como um instrumento auxiliar do ensino.

2. *A quem se dirigem?*

- à população escolar que nada ou muito pouco sabe de Basic e pretende acompanhar a introdução das novas tecnologias no Ensino.

3. *Quem são os monitores?*

- professores ou estudantes que para o efeito se ofereçam.

4. *Metodologia*

- Os cursos organizam-se em 6 sessões de 2 horas cada, distribuídas ao longo de 3 semanas.
- Cada microcomputador admite no máximo 3 utilizadores e 1 monitor.
- As sessões compreendem:
 - utilização/manuseamento do teclado;
 - análise de programas já construídos e principais instruções de Basic;
 - problemas para equacionar;
 - construção de pequenos programas pelo utilizador.



Fig. 18

5. Orientação das sessões dos cursos

- Embora os monitores tenham toda a liberdade para modificar a ordem e mesmo o conteúdo das sessões, convém definir uma orientação para que o curso tenha um fio condutor e não se afaste dos objectivos.

Devem ter-se em conta duas questões:

- só se aprende pensando e fazendo;
- ensinar muito (muitas instruções), não significa necessariamente aprender muito.

Propostas de distribuição de conteúdos pelas sessões

- 1.^a sessão — Apresentação. Objectivos do curso. As várias utilizações possíveis do micro (calculadora, jogos, construção de programas).
 - Cassete de apresentação do Spectrum (prática com o teclado; algumas instruções e seu significado).
 - O que é um programa? Sequência de instruções. Diferença entre numerar e não numerar as instruções.
- 2.^a sessão — Análise de programas com a utilização das instruções: REM, LET, PRINT, STOP, INPUT, GO TO, CLS.
 - Variáveis numéricas. Emendar erros ou apagar. Instruções DELETE, EDIT, NEW.
- 3.^a sessão — Análise de programas com a utilização das instruções: FOR, TO, STEP, NEXT, IF...THEN, INT, RND, RANDOM, PAUSE.
 - Variáveis alfanuméricas. Referência à instrução BEEP.
- 4.^a sessão — Análise de programas com a utilização das instruções e funções: DEF FN, VAL, SIN, COS, TAN, ABS.
 - Referência/exemplificação com as instruções FLASH, BRIGHT, BORDER, PAPER, INK.
- 5.^a sessão — Análise de programas com a utilização das instruções: PLOT, DRAW, CIRCLE.
 - Construção de pequenos programas (por ex.: sistema de 2 eixos ortogonais, um boneco, um cone de circunferências, uma grelha de avaliação). Gráficos de funções.
- 6.^a sessão — Análise de programas com a utilização das instruções: GO SUB, RETURN, READ, DATA, RESTORE.
 - Símbolos gráficos definidos pelo utilizador.
 - Gravação de um programa (instruções SAVE e VERIFY).

NOTAS: 1. Em cada sessão os monitores devem propor 1 ou 2 problemas para equacionar e programar, tendo em conta as instruções já conhecidas.

2. Os programas a ser analisados deverão ser muito simples no início, vindo a crescer de grau de complexidade com o

decorrer das sessões. Para o efeito encontram-se à disposição dos monitores umas folhas da Escola Secundária de St. André, do Barreiro, e alguns módulos utilizados na cadeira de Metodologia da Matemática da Faculdade de Ciências de Lisboa.

NOTA FINAL: Os professores e estudantes que participem nos cursos devem colaborar posteriormente com a Secção, no apoio aos próximos cursos, sala de tempos livres, às respectivas turmas (alunos) ou Grupos Pedagógicos (professores) e em iniciativas da secção.

Pela simples razão de o grupo dinamizador da secção de informática ser o de matemática, decidiram utilizar como textos de apoio, entre outros, alguns módulos para o efeito preparados pelo Departamento de Educação da Faculdade de Ciências de Lisboa.

De qualquer modo, e mesmo para outras disciplinas — ou melhor, para a utilização do computador em geral — é de notar a planificação de cada módulo, que se destina, no essencial, a apresentar e explicar, com aplicações práticas, os comandos do computador.

MÓDULO 1: Instruções básicas

```
5 REM soma dos naturais pares
de 1 a n
10 LET s=0
20 INPUT "Soma dos numeros nat
urais pares de 1 ate ?";n
30 FOR i=2 TO n STEP 2
40 LET s=s+i
50 NEXT i
60 PRINT "Soma dos numeros nat
urais pares de dois ate ";n;" =
";s
70 INPUT "Novo calculo ? s/n "
;x$
80 IF x$="s" OR x$="S" THEN CL
S : GO TO 10
90 STOP
```

REM — instrução não executável; destina-se à escrita de comentários ao programa.

LET — definição de variáveis.

INPUT — o computador pergunta pelo valor de uma variável.

FOR...NEXT — ciclo; todas as instruções entre FOR e NEXT são executadas para os valores da variável de ciclo (neste caso i) entre o valor inicial e o valor final, com passo definido.

PRINT — instrução de saída de resultados e/ou escrita de texto.

IF...THEN — instrução de decisão; se a condição se verifica, o

computador executa as instruções que seguem o THEN (na mesma linha); em caso contrário passa à linha seguinte.

GO TO — remete a execução do programa para a instrução indicada.
CLS — limpa o écran.

OBSERVAÇÕES

- linha 30 — quando o passo é 1 pode omitir-se STEP 1.
- linha 10 — inicialização da variável s, dado que esta variável é utilizada na linha 40 (atribuindo a s o valor s+i) tem de estar definida previamente.
- linha 80 — pode substituir a sequência CLS:GO TO 10 por RUN (a instrução RUN limpa as variáveis dos seus valores presentes).
- linha 90 — a instrução STOP no final do programa é dispensável.
- linha 70 — as variáveis alfanuméricas (cadeias de letras e/ou números) são definidas através de \$ (no programa presente x\$).

MÓDULO 2: Funções

Programa B

```

5 REM função
10 DEF FN f(x)=ABS SIN x
20 FOR i=0 TO 2*PI STEP PI/3
30 PRINT i, FN f(i)
40 NEXT i
50 STOP

5 REM simulacao do lancamento
de uma moeda
10 LET c=0
20 INPUT "Quantos lancamentos
?"; n
30 FOR j=1 TO n
40 LET y=INT (10*RND)
50 IF y>4 THEN LET c=c+1
60 NEXT j
70 PRINT n; " Lancamentos " c; "
Caras " n-c; " Coroadas "
80 STOP

```

DEF FN — permite definir uma função pelo utilizador (neste caso f), que é em todo o programa chamada através de FN.

SIN, COS, TAN — funções pré-definidas no próprio computador (notar que os seus argumentos devem ser introduzidos em radianos).

ABS — valor absoluto.

INT — parte inteira de um real (característica).

RND — gera números (pseudo) aleatórios no intervalo [0,1].

OBSERVAÇÕES

- no programa B, o ciclo que vai da linha 30 à linha 60 gera um inteiro aleatório $0 < y < 9$;
- faz o teste para distribuição da probabilidade incrementando I à variável c (contador de caras) quando for caso disso;
- o símbolo ' (linha 70 do programa B) é utilizado para fazer saltar para a linha seguinte na saída;
- a variável PI está definida no Spectrum com o valor (aproximado) de 3.1415927

MÓDULO 3: gráficos de alta resolução

```

10 PLOT 70,70
20 DRAW 0,30: DRAW 30,0: DRAW
0,-30: DRAW -30,0
30 DRAW 30,30
40 CIRCLE 85,85,22
50 STOP

```

```

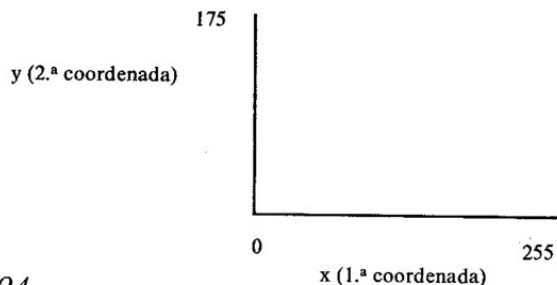
5 REM chuva
10 PLOT 102,70
20 DRAW 0,30: DRAW -2*PI/3
30 PLOT 142,95
40 DRAW 0,-60
50 DRAW -10,0,-PI
60 FOR m=0 TO 255 STEP 5: PLOT
m,60
80 DRAW INK 5;0,90
90 NEXT m

```

- PLOT — desenha um pixel (ponto) nas coordenadas dadas
- DRAW — desenha um segmento de recta (a partir do PLOT anterior). Exemplo: DRAW 7,3
- CIRCLE — desenha uma circunferência. Exemplo: CIRCLE 85,85,22 desenha uma circunferência de centro (85,85) e raio 22
- DRAW x,y,z — executa uma rotação (a partir do PLOT anterior) terminando em (x,y) e com medida z (em radianos)

OBSERVAÇÕES

- caso não esteja definido PLOT no programa, o computador atribui-lhe as coordenadas (0,0)
- o sistema de coordenadas de alta resolução está representado na figura:



MÓDULO 4

```

10 REM calculo do quadrado de
um numero
20 PRINT "Calculo do quadrado
de um numero"
30 INPUT "Numero:"; x
40 IF x=0 THEN GO SUB 100: GO
TO 60
50 LET y=x^2
60 PRINT "x;" ao quadrado = ";
y
70 STOP
100 REM rotina
110 LET y=x*x
120 RETURN

```

GO SUB m — remete o programa para a execução da linha m até à instrução RETURN, após o que prossegue a execução na instrução consecutiva ao GO SUB m

RETURN — termina a execução da Subrotina

READ — executa a leitura dos dados que são colocados na instrução DATA

DIM — pré-defina as variáveis indexadas que o programa vai utilizar. Exemplo: DIMx(50) pré-defina a variável x(i), limitando esta família a 50 elementos

OBSERVAÇÕES

— notar que na linha 60 a instrução STOP é indispensável

— os dados são separados por vírgulas; no caso de serem alfanuméricos deverão ser colocados entre "

Pelo inegável interesse que possui — atendendo à simplicidade, aliada à variedade — apresenta-se seguidamente o conjunto de rotinas e programas utilizados pelo Núcleo de Estágio da Escola Secundária de Santo André (Barreiro), para o Curso de Basic.

★

Sumário I: instruções REM, LET, PRINT, STOP, INPUT, PAUSE, e GOTO.

```

5 REM soma de dois numeros
10 LET a=832
20 LET b=-1247
30 LET s=a+b
40 PRINT a,b,s
50 STOP

```

```

5 REM soma de dois numeros
10 LET a=832
20 LET b=-1247
30 LET s=a+b
40 PRINT a,b,s
50 STOP

```

```

5 REM soma de dois numeros
10 LET a=832
20 LET b=-1247
30 LET s=a+b
40 PRINT AT 0,0;a;AT 0,25;b;AT
14,14;s
50 STOP

```

```

5 REM soma de dois numeros
10 LET a=832
20 LET b=-1247
30 LET s=a+b
40 PRINT "a=";a: PRINT "b=";b:
PRINT "s=";s
50 STOP

```

```

5 REM soma de dois numeros
10 INPUT a
20 INPUT b
30 LET s=a+b
40 PRINT "a=";a: PRINT "b=";b:
PRINT "s=";s
50 STOP

```

```

5 REM soma de dois numeros
10 INPUT "a=";a
20 INPUT "b=";b
30 LET s=a+b
40 PRINT "a=";a: PRINT "b=";b:
PRINT "s=";s
50 STOP

```

```

5 REM calculo do valor dum a e
xpressão designatoria para concr

```



```

etizacoes de varaveis
5 PRINT AT 0,8;"(a-b↑2)/(b+a↑
2)"
10 INPUT "a=";a
20 INPUT "b=";b
30 LET v=(a-b↑2)/(a+b↑2)
40 PRINT "a=";a; PRINT "b=";b;
PRINT "v=";v
50 STOP

```

```

5 REM calculo do valor duma e
xpressão designatoria para concr
etizacoes de varaveis
5 PRINT AT 0,8;"(a-b↑2)/(b+a↑
2)"
10 INPUT "a=";a
20 INPUT "b=";b
30 LET v=(a-b↑2)/(a+b↑2)
40 PRINT "a=";a; PRINT "b=";b;
PRINT "v=";v
50 STOP

```

```

5 REM conversa de temperatura
s
10 INPUT "Graus Fahrenheit F =
";f
20 LET c=(f-32)*5/9
30 PRINT "Graus centigrados C=
";c
40 PAUSE 200
50 CLS : GO TO 10

```

```

5 REM conversa de temperatura
s
10 INPUT "Graus centigrados C=
";c
20 LET f=(9*c+160)/5
30 PRINT "Graus Fahrenheit F =
";f
40 PAUSE 200
50 CLS : GO TO 10

```

- Exercícios: 1 — Construir programas para a determinação de áreas e volumes de sólidos;
- 2 — Construir um programa para calcular os vencimentos dos trabalhadores de uma loja em que se descontam 19% para a Caixa e, sobre o restante, 12% para a reforma. Escalão A — 30 000\$00; escalão B — 22 000\$00.

★

Sumário 2: comandos IF... THEN, FLASH, FOR... TO... STEP, NEXT, INT e GRAPHICS.
variáveis alfanuméricas: A\$

```

5 REM apresentacao
10 PRINT "Olá! EU sou o SPECTRU
M! E tu, como te chamas?"
20 INPUT a$
30 CLS
40 PRINT "Estou a gramar-te ";
a$; "!"
50 PAUSE 200
60 CLS
70 GO TO 10

```

```

5 REM Saio com guarda chuva o
u não?
10 PRINT "Esta a chover ?(s/n)
"
20 INPUT a$
25 CLS
30 IF a$="s" THEN GO TO 70
40 PRINT AT 10,2;"Como não cho
ve, não preciso do guarda-chuva."
: STOP
70 PRINT AT 10,2;"Como esta
a chover , levo o guarda-chuva"

```

```

5 REM Divisibilidade - Verifi
car se um número A é múltiplo de
um número B.
10 INPUT "A = ";A; PRINT "A ="
;A
20 INPUT "B = ";B; PRINT "B ="
;B
30 LET q=a/b
40 IF q=INT (q) THEN PRINT AT
10,0; FLASH 1;A;" é múltiplo de
";B; GO TO 50
50 PRINT AT 10,0; FLASH 1;A;"
não é múltiplo de ";B
60 PAUSE 200
65 CLS
70 PRINT "QUER CONTINUAR? (s/n
)"
80 INPUT r$
90 IF r$="s" THEN CLS : GO TO
10
100 CLS : PRINT AT 10,11; BRIGH
T 1;" F I M "

```

```

5 REM quadrados perfeitos
10 INPUT "n=";n
20 FOR i=1 TO n
30 PRINT i,i↑2
40 NEXT i

```

```

5 REM raizes quadradas dos nu
meros de 1 ate n
10 INPUT "n = ";n
20 FOR i=1 TO n
30 PRINT i,SQR i
40 NEXT i

```

```

5 REM raizes quadradas inteir
os dos numeros de 1 a n
10 INPUT "n = ";n
20 FOR i=1 TO n
30 PRINT i,INT (SQR i)
40 NEXT i

```

```

5 REM numeros primos menores
que um dado N
10 INPUT "N = ";N
20 FOR p=2 TO N
30 FOR r=2 TO SQR (p)
40 LET q=p/r
50 IF q=INT (q) THEN GO TO 80
60 NEXT r
70 PRINT p
80 NEXT p

```

```

5 REM numeros primos menores
que um dado N
10 INPUT "N = ";N
15 PRINT 2,3
20 FOR p=5 TO N STEP 2
30 FOR r=3 TO SQR (p) STEP 2
40 LET q=p/r
50 IF q=INT (q) THEN GO TO 80
60 NEXT r
70 PRINT p
80 NEXT p

```

```

5 REM Multiplos pares de 7
10 INPUT "N = ";N
20 FOR i=0 TO n STEP 2
30 PRINT 7*i
40 NEXT i

```

```

5 REM Decomposicao em factore
s primos
10 INPUT x
15 PRINT ; FLASH 1;x: PRINT
20 LET y=1

```

```

30 LET y=y+1
40 IF x/y=INT (x/y) THEN GO TO
100

```

```

50 GO TO 30
100 PRINT y
110 LET x=x/y
115 IF x=1 THEN STOP
120 GO TO 40

```

```

5 REM maximo divisor comum
10 INPUT x,y
15 PRINT "m.d.c. (";x;",";y;")
= "; FLASH 1;"2";
30 IF x>y THEN GO SUB 200
35 LET z=x
40 IF y/z=INT (y/z) THEN GO TO
80
50 LET z=z-1
60 GO TO 40
80 IF x/z=INT (x/z) THEN GO TO
120
90 GO TO 50
120 PRINT CHR$ 8;z: STOP
200 LET c=x: LET x=y: LET y=c
210 RETURN

```

```

5 REM desenho
10 FOR n=1 TO 20
20 PRINT AT n,n;"███"
30 NEXT n

```

```

5 REM desenho
10 FOR n=0 TO 21 STEP 3
15 FOR p=30 TO 0 STEP -3
20 PRINT AT n,p;"███"
25 NEXT p
30 NEXT n

```

- Exercícios: 1 — Construir um programa para calcular o quociente inteiro de dois números A e B.
 2 — Fazer organigramas dos programas apresentados.
 3 — Construir um programa para determinar o M. M. C. de dois números.

★

Sumário 3: comandos PLOT, DRAW e CIRCLE.

```

5 REM ciclos de plots
10 FOR n=0 TO 175
20 PLOT 0,n
30 NEXT n
40 FOR n=0 TO 255
50 PLOT n,0
60 NEXT n
70 STOP

```

```

1 REM Quadricula
10 LET n=0
20 FOR m=0 TO 170 STEP 10
30 PLOT n,m: DRAW n+250,0
40 NEXT m
50 LET m=0
60 FOR n=0 TO 250 STEP 10
70 PLOT n,m: DRAW 0,m+170
80 NEXT n

```

```

1 REM circunferencias
10 FOR n=1 TO 80 STEP 5
20 CIRCLE 125,80,n
30 CIRCLE 125,80,80-n
40 NEXT n

```

```

1 REM circunferencias
10 FOR n=1 TO 80 STEP 5
20 CIRCLE 170,80,n
30 CIRCLE n,n,n
40 NEXT n

```

```

5 REM cone
10 LET a=1: LET b=1
20 FOR t=1 TO 185 STEP 5
30 CIRCLE t,a,b
40 LET a=a+2: LET b=b+2
50 NEXT t

```

```

10 PRINT "REDE"
30 INPUT "Grossura 1-25 (1 Gro
ss - 25 Fino) ";b
40 LET a=0
50 PLOT 0,175: DRAW 0,-175
60 DRAW 175,0
70 FOR t=175 TO 0 STEP -b
80 PLOT 0,t: DRAW a,-t
90 LET a=a+b
100 NEXT t
110 STOP

```

```

10 FOR n=1 TO 10
20 FOR j=1 TO 7 STEP .1
30 PLOT n*j-COS (j)+110,n*j*SI
N (j)+100
40 NEXT j
50 NEXT n

```

```

5 REM orientacao duma recta
10 INPUT "x1=";x1,"y1=";y1
20 INPUT "x2=";x2,"y2=";y2
30 FOR n=0 TO 255: PLOT n,0: P

```

```

LOT n,175: NEXT n
24 FOR n=0 TO 175: PLOT 0,n: P
LOT 255,n: NEXT n
30 LET m=(y2-y1)/(x2-x1)
40 FOR n=0 TO ABS (x2-x1) STEP
.05
50 PLOT x1+n,(y1+n)*m
60 NEXT n
70 STOP

```

- Exercícios: 1 — Construir um sistema de dois eixos ortogonais, um vertical e outro horizontal, com origem no centro do écran.
- 2 — Construir um programa de desenho de figuras geométricas planas. (Sugestão: construir um triângulo do qual se conhecem os vértices.)

★

Sumário 4: funções DEF FN, FN e VAL.

```

5 REM gerador de numeros pare
s de 1 ate 1000
10 DEF FN p(x)=2*x
20 FOR x=1 TO 500 STEP 1
30 PRINT FN p(x)
40 NEXT x
50 STOP
105 REM gerador de numeros pare
s de 1 ate 1000
110 LET p$="2*x"
120 FOR x=1 TO 500
130 PRINT VAL (p$)
140 NEXT x
150 STOP
505 REM gerador de numeros pare
s de 1 ate 1000
510 FOR n=2 TO 1000 STEP 2
520 PRINT n
530 NEXT n
540 STOP

```

```

5 REM funcao
10 DEF FN p(x)=x^2+1
20 FOR x=0 TO 10 STEP 1
30 PRINT x,FN p(x)
40 NEXT x

```

```

5 REM funcoes
10 LET p$="x^2+1"
20 FOR x=0 TO 10 STEP 1
30 PRINT x,VAL p$
40 NEXT x

```

```

5 REM funcoes
10 PRINT "UTILIZE UMA FUNCAO
DE VARIÁVEL 'X'."
20 INPUT P$: CLS
30 FOR X=0 TO 10 STEP .5
40 PRINT X, VAL (P$)
50 NEXT X

```

```

5 REM equacao de segundo grau
7 PRINT "a*x^2+b*x+c=0"
10 INPUT "a="; a: INPUT "b="; b:
INPUT "c="; c
30 LET d=b*b-4*a*c
40 LET x1=(-b+SQR (d))/2*a
50 LET x2=(-b-SQR (d))/2*a
60 PRINT AT 6,0;"x1=";x1,"x2="
;x2

```

```

5 REM equacao de segundo grau
10 PRINT "Calculo dos zeros
do polinómio de segundo grau"
15 INPUT "a="; a: INPUT "b="; b:
INPUT "c="; c
20 PRINT a;"*x^2+"; b;"*x+"; c
30 LET d=b*b-4*a*c
40 LET x1=(-b+SQR (d))/2*a
50 LET x2=(-b-SQR (d))/2*a
60 PRINT AT 6,0;"x1=";x1,"x2="
;x2

```

- Exercícios: 1 — Construir um programa que permita fazer translações. (Sugestões: 0(0,0) e 0'(a,b) em que (a,b) é a origem do sistema de eixos do exercício do sumário 3.)
- 2 — Construir um programa para desenhar gráficos de funções. (Sugestão: recorde o exercício do sistema de eixos e atenda ao exercício 1.)

2.2 — A programação

Numa certa fase inicial, o computador foi encarado — pela maioria das pessoas — como uma espécie de máquina de jogos vídeo, muito mais desenvolvida do que as até então existentes. E não deixa de ser interessante notar que mesmo nos grandes computadores profissionais, a cassette (ou diskette) de demonstração continha sempre pelo menos um jogo — mais ou menos complexo, consoante as capacidades do equipamento.

Nessa perspectiva — e porque existem no mercado milhares de programas comerciais prontos a ser utilizados sem mais trabalho que o da sua introdução nos micros — ainda hoje muita gente utiliza os computadores apenas com programação adquirida. O que, entre outras coisas, constitui um enorme desperdício das capacidades dos mesmos.

Uma das grandes possibilidades surgidas na concepção básica dos computadores é precisamente a de se tratar de máquinas que, embora fabricadas em série, permitem uma utilização absolutamente individualizada, com aplicações específicas a cada caso. Se assim não fosse, aliás, pouco mais teríamos do que máquinas de calcular, com algumas capacidades extra, mas que executariam basicamente o mesmo e limitado tipo de tarefas (cálculos).

Ao contrário, precisamente, da máquina de calcular, o computador possui o tal vocabulário e sintaxe já referidos, permitindo portanto um número infinito de combinações, que resultam em outras tantas tarefas. Na prática é mesmo impossível determinar a totalidade de possibilidades de trabalho de um computador.

Para que o computador execute qualquer dessas tarefas é necessário fornecer-lhe a instrução correspondente. Por exemplo, se se quiser que ele imprima o número «1» no écran:

PRINT 1

Trata-se de uma instrução simples. Todavia, se se quiser que ele imprima esse número num ponto escolhido do écran:

PRINT AT 10,10; «1»

porque, mesmo pensando em termos puramente humanos, já se não trata de uma instrução simples. Na realidade é bastante complexa. Vejamos:

Procurando estabelecer um paralelo com o ser humano, uma frase do tipo «Escreva um 1 num quadro, mas com um afastamento de 10 centímetros na horizontal e outro de 10 centímetros na vertical (coordenadas)» envolve vários cálculos e instruções:

- 1 — Escreva o número «1»;
- 2 — Num quadro (poderia ser num caderno, etc.);
- 3 — Com a seguinte localização:

- 3.1 — a 10 centímetros do lado esquerdo do quadro;
- 3.2 — a 10 centímetros do topo do quadro.

Embora inicialmente não pareça, o simples acto de executar tal tarefa implica, como se vê, algumas instruções e cálculos.

Não será de estranhar que o mesmo seja necessário para o computador — nunca é demais sublinhá-lo —: *extensão humana*. Na realidade, porém, as instruções para o computador encontram-se simplificadas. Por exemplo:

PRINT — para, «imprima no écran do televisor»
 LPRINT — para, «imprima no papel da impressora»
 CLS — para, «apague todo o écran do televisor»

Talvez seja agora mais fácil entender o que é um *programa*. Não passa de um conjunto de instruções (ordens) destinadas a levar o computador a executar determinado tipo de tarefas e cálculos, com a intenção de atingir um objectivo final previsto ou desejado.

Programar implica, portanto, planificar. E planificar não é possível sem analisar. Há, sem dúvida, muitos métodos de programação, dependendo não apenas do tipo de tarefas que se pretende ver executadas, como ainda do próprio programador (tal como em literatura, cada programador tem um estilo próprio, perfeitamente identificável após a elaboração de alguns programas). Todavia, não deixa de ser importante — sobretudo para quem está a começar — indicar algumas regras de base, que orientem o trabalho na sua metodologia mais genérica.

Os passos da programação

Simple ou complexo, o acto de programar implica *sempre* — ainda que não sistematizados — os mesmo passos, ou seja, as mesmas necessidades de análise do problema, projecção do mesmo, sua realização progressiva, teste e, por fim, manutenção.

A análise

Posto um qualquer problema destinado a ser solucionado por meios informáticos, importa analisá-lo, antes mesmo de iniciar quaisquer linhas de programa.

Analisar é precisamente decompor um problema nas suas diferentes partes, por forma a permitir projectar várias rotinas que no seu conjunto o solucionem.

A primeira coisa a fazer é, *sempre*, definir qual o *objectivo* final do trabalho do computador; ou seja, qual o *resultado* que se pretende obter (total de uma soma, desenho de uma figura geométrica, execução de uma ária musical, etc.).

Seguidamente, procurar definir, ainda que genericamente, quais as rotinas e variáveis necessárias para o efeito, quais os comandos-base indispensáveis (para o caso do som, por exemplo, o comando BEEP no Spectrum, SOUND no TC 2068, etc.). Se possível, procurar, ainda nesta fase, determinar aproximadamente o montante de memória necessário ao programa e sua execução — não há nada pior que começar trabalhosa e desenvolver um «grande» projecto que se torna inteiramente irrealizável, por razões básicas de insuficiência de espaço na memória do computador.

O projecto

Após a análise, começar por definir, com a maior exactidão possível, qual o *ponto de partida* para atingir o resultado previsto (o que nem

sempre é fácil, uma vez que, no posterior desenvolvimento do programa, acaba por verificar-se a necessidade — ou utilidade — de o iniciar de outro modo; por essa razão, nunca deve atribuir-se à primeira linha de um programa o número «1», nem mesmo o «10». É uma boa regra começar o programa com a linha número «100»).

Definido o ponto de partida — e tendo, da análise anterior, o problema decomposto em partes — esboçar um organigrama, a partir dessas mesmas partes:

O organigrama (Fig. 19) permite, como é evidente, visualizar de forma organizada o encadeamento lógico das operações a efectuar, bem como as respectivas ramificações e ligações entre elas — o que, só por si, constitui preciosa ajuda, até na indicação dos comandos a utilizar em algumas operações.

Esta é uma boa altura para confirmar a correcção da análise (procedendo a alterações, se for caso disso), bem como a questão da memória necessária. É igualmente a altura de definir as variáveis a serem utilizadas pelo programa.

Antes ainda de iniciar a programação propriamente dita, há que procurar determinar quais os elementos que terão que ser fornecidos ao computador — quer no acto da programação, quer no decorrer da execução do programa — para que ele possa desempenhar o seu trabalho. (Também aqui se verifica, posteriormente, a necessidade de fornecer mais elementos, ou mesmo alterar premissas inicialmente julgadas necessárias. Por tal razão se deve considerar como regra de programação o estabelecimento de um intervalo de pelo menos 10 linhas entre cada instrução. Se se começar na linha 100, a segunda será a 110, a terceira a 120, e assim por diante).

A realização do programa

Finalmente, iniciar a elaboração do programa. Ou seja, ir compondo linhas com as instruções que se vão tornando necessárias para que o computador possa, progressivamente, ir realizando as suas tarefas.

Já atrás indicámos duas regras-base (o início na linha 100 e o intervalo de 10 entre cada linha) que permitirão uma maior elasticidade e segurança no trabalho. Uma terceira, permitirá, para além das vantagens indicadas, não só um acesso rápido e organizado a qualquer das partes do problema (ver organigrama), como também guardar blocos com rotinas que poderão servir, posteriormente, para inserir em qualquer outro programa. Trata-se daquilo a que se chama o método de *programação modular* (um breve exemplo disso foi já visto no texto de apoio da Faculdade de Ciências de Lisboa, anteriormente apresentado).

O «segredo» é bem simples: elaborar, bloco a bloco, o conjunto de instruções que irão servir para executar as operações necessárias a uma determinada *parte* do programa. Caso o programa não seja muito grande, é uma boa regra iniciar cada bloco num milhar (1.º bloco na linha 1000, segundo na linha 2000, e assim por diante). Se for grande, poderá recor-

rer-se a inícios de blocos na casa das centenas (1.º bloco na linha 500, 2.º na linha 1000, etc.).

Qualquer destas rotinas poderá ser chamada, em qualquer momento e com muita rapidez, com o comando GOSUB.

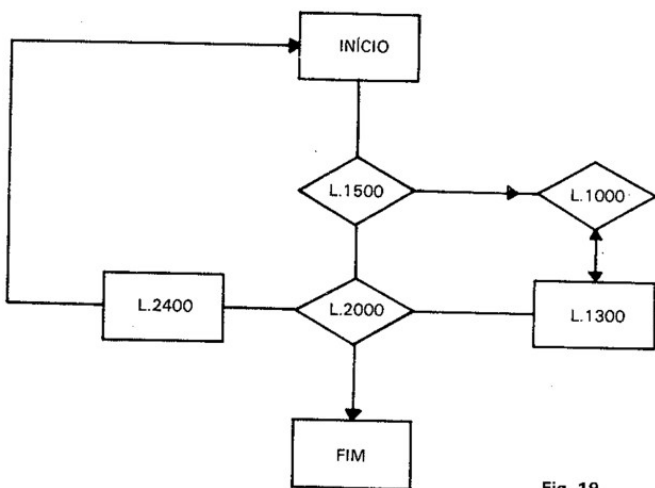


Fig. 19

Um pequeno exemplo

Vamos supor que pretendo que o computador, face a valores indicados pelo operador, trace um círculo, com um raio determinado, num certo ponto do écran.

Uma vez que o ZX Spectrum (e o TC 2068 e o TS 1500) possuem o comando CIRCLE, a tarefa encontra-se simplificada. Mesmo assim, não basta fazer um programa com a linha:

```
100 CIRCLE 5, 3, 10
```

porque tal programa se limitaria a traçar um círculo de raio 10 nas coordenadas 5 e 3 do écran.

O problema, tal como foi apresentado, implica que o operador possa indicar os valores. Isso obriga a iniciar o programa com uma instrução INPUT, que diz ao computador para aguardar a introdução de dados a partir do teclado:

```
100 INPUT x
```

por exemplo.

Todavia, se introduzirmos esta linha assim, o programa só vai servir para ser utilizado por quem o elaborou. Porque ao começar, o que aparece no écran é apenas o que representa a figura seguinte.

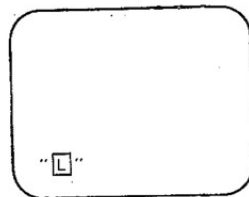


Fig. 20

Para que o programa sirva a qualquer operador, deve começar por fornecer alguma informação. Razão que leva a programar a primeira linha

```
100 INPUT "Qual a coordenada de x?"; x
```

O mesmo para a coordenada de y, bem como para o raio. O que nos dará, como resultado, um programa

```
100 INPUT "Qual a coordenada de x?"; x
110 INPUT "Qual a coordenada de y?"; y
120 INPUT "Qual o raio do círculo?"; r
130 CIRCLE x, y, r
```

Após o programa executado, o computador pára, tendo desenhado um círculo com o raio indicado, com o centro no encontro das coordenadas x e y.

O programa está assim concluído. Todavia, ao experimentá-lo, podemos verificar que, com pouco mais instruções, pode ser bastante melhorado — *sem deixar de servir para cumprir o objectivo inicial*. Por exemplo, pode não parar após o desenho do círculo, perguntando ao operador se quer desenhar outro. Nesse caso, basta acrescentar

```
140 INPUT "Quer desenhar outro?"; i$
150 IF i$="s" OR i$="S" THEN GOTO 100
```

Enfim, a partir daqui, como já atrás disse, o desenvolvimento do programa torna-se ilimitado, dependendo do grau de aperfeiçoamento e opções do programador.

De qualquer modo, e após cumpridas as regras básicas atrás indicadas, o número e tipo das tarefas a executar até atingir o objectivo final irá sendo detectado através da experimentação ocorrida durante o desenvolvimento do programa. Só a simulação e a verificação dos resultados permitirá detectar erros e aperfeiçoar resultados.

Quando o programador termina (ou pensa que terminou) o programa e verifica que o mesmo funciona, atingindo o objectivo previsto, não deve pensar que o mesmo se encontra pronto. Talvez que para ele que o projectou e elaborou e, portanto, tem, de algum modo, o seu cérebro em sintonia com o mesmo, isso seja verdade. Todavia, um programa realiza-se para ser utilizado por outras pessoas que, do mesmo, apenas sabem que serve para executar tal ou tal trabalho. Assim, é necessário testá-lo, para verificar como responde a formas de utilização não previstas.

É a fase de *teste e afinação* do programa. E para o fazer do melhor modo possível nada existe de melhor que dá-lo a utilizar a outros utentes, verificando, durante essa utilização, quais as deficiências notadas e corrigindo, pontualmente, o que for necessário.

Manutenção

Aquilo a que na informática se chama «manutenção» é, ao fim e ao cabo e com maior propriedade, um processo não finito de *adaptações*. Ou seja, pronto o programa, testado e verificando-se que o mesmo responde correctamente a todas as situações possíveis, sem erros, ainda assim se não pode considerar fechado. Na verdade, um programa de computador é, sempre, uma obra aberta — pronto a ser parcialmente adaptado a necessidades específicas deste ou daquele trabalho, deste ou daquele utilizador. Isso mesmo ocorre nos programas profissionais.

Por essa razão, o programador deve estar disponível para, pacientemente, introduzir essas alterações, a solicitação dos utilizadores, não devendo *já* considerar tais solicitações como críticas destrutivas ao seu trabalho, mas antes como reconhecimento do valor do mesmo, que pode, no entanto, ser *melhorado*. Em alternativa, poderá fornecer instruções detalhadas sobre o programa, o que permitirá ao utilizador efectuar ele próprio as modificações que entender. Isto mesmo é o que fazem algumas firmas, mesmo relativamente a programas comerciais (exemplo: a Tasman relativamente a programas como o Tasword).

Breve referência à programação em código máquina

Neste capítulo tive fundamentalmente a preocupação de introduzir, de modo muito simples e acessível, os conceitos de programa e programação — conceitos esses que passam, numa primeira fase, pela linguagem BASIC, pelas razões apontadas no decorrer do texto.

É evidente que, sobretudo quando se pretende tirar total rendimento do computador em programas aplicados a certas áreas do ensino, a linguagem máquina (a verdadeira linguagem do microcomputador) é a preferível, uma vez que aumenta de uma forma espantosa as potencialidades do sistema. Deste modo, aprender a programar em código máquina

deverá ser inevitavelmente uma meta a atingir por quem pretende executar trabalho sério e profundo.

Contudo — e embora no fundo seja até mais simples que a programação em BASIC — dominar a programação máquina implica readquirir conceitos, reaprender vocabulário e sintaxe (bem mais simples, insisto), em suma, visualizar o trabalho conceptual de modo diverso. Ora, como é óbvio, tais alterações implicam todo um processo de introdução e desenvolvimento, o que constituiria, por si só, uma nova obra, pelo menos tão extensa como esta. Além do que, para enveredar por essa fase de trabalho com computadores, o mínimo exigível é que se tenham conhecimentos razoáveis do BASIC.

Assim sendo, esta referência servirá tão só para aconselhar a passagem a esse tipo de programação, logo que o utilizador se sintá à vontade no BASIC. De qualquer modo — e para que, os que já programam em linguagem máquina, não se sintam inteiramente frustrados — o programa LIFE, para as Ciências Naturais, utiliza o código máquina (facto que não impede, pelo modo como está concebido, que qualquer um possa introduzi-lo no computador).

Além disso — e sobretudo para aqueles que ainda se não iniciaram nesta linguagem (embora possa ter igualmente muita utilidade para os outros) — aqui fica um curto programa, muito útil, destinado a permitir a introdução de programas em código no computador. Basta teclá-lo e indicar-lhe qual o primeiro endereço, para que ele vá introduzindo automaticamente o código, a partir de listagem hexadecimal, ou decimal, a partir de grupos de dois dígitos no primeiro caso, (Exemplo: 0F 45 01 EF 3C ...), ou números compreendidos entre 0 e 255, no segundo.

```

10 PRINT "**Carregador de codi
90-maquina**"
11 POKE 23658,8
12 PRINT AT 5,3
13 PRINT " 1-HEXADECIMAL"
14 PRINT " 2-DECIMAL"
15 INPUT "OPCAO (1/2)";o$
16 IF CODE o$<49 OR CODE o$>50
THEN GO TO 15
20 LET o=VAL o$
22 IF LEN o$<>1 THEN GO TO 15
30 INPUT "Endereco inicial?"
:END
32 LET INIC=end
38 CLS : PRINT "Prima 'FIM' pa
ra terminar"
40 PRINT "Indique codigo um p
or um"
45 IF o=2 THEN GO TO 1500
50 PRINT "sempre em maiusculas"
55 PRINT "ENDER. HEX DEC"
60 PRINT end;" = ";
70 INPUT a$
80 IF a$="FIM" THEN GO TO 1000
85 IF LEN a$<>2 THEN GO TO 120
90 LET al=CODE a$(1)-48

```



```

100 IF al>9 THEN LET al=al-7
110 LET ns=10*al
120 LET ba=CODE a$(2)-48
130 IF ba>9 THEN LET ba=ba-7
140 LET ns=ns+ba
150 PRINT a$;" ";ns
160 POKE end,ns
170 LET end=end+1
180 GO TO 80
1000 LET TER=end
1010 LET COMP=TER-INIC
1020 INPUT "Quer gravar o código
?($/n)";i$
1030 IF i$="N" THEN GO TO 1100
1040 CLS:PRINT "Introduza a ca
ssette respectiva"
1050 INPUT "Nome do programa-";n$
1060 PRINT " ";n$
1070 INPUT "Esta correcto?($/n)";i$
1080 IF i$="N" THEN GO TO 1050
1090 SAVE n$CODE INIC,COMP
1100 CLS:PRINT "AT 21,0;"T
ransferencia terminada";STOP
1200 PRINT #0;AT 0,0;"TEM QUE SE
R DOIS DIGITOS":PAUSE 50:GO TO
70
1500 CLS:PRINT "ENDER. DEC"
1502 PRINT end;"=";
1505 INPUT "Código:";i$
1507 IF i$="FIM" THEN GO TO 1000
1510 IF VAL i$>255 OR VAL i$<0 T
HEN GO TO 1505
1530 LET ns=VAL i$
1540 PRINT ns
1500 GO TO 1502

```

2.3 — Aulas

Como, também erradamente poderá pensar-se, utilizar o computador nas aulas não significa transferir para aquele o papel do professor. Isso seria, de resto, impossível, a menos que se pretendessem aulas repetitivas, em que o imprevisto e a resposta pontual a determinadas situações fossem banidas. O que é fácil de entender, atendendo ao que acabou de expôr-se acerca da programação.

O que pode fazer-se — e com resultados, por vezes, espectaculares — é utilizar o computador como auxiliar, em maior ou menor grau, ou seja, transferindo para ele maior ou menor tempo de intervenção, com maior ou menor quantidade de informação a fornecer. E, sobretudo, utilizando-o como substituto vantajoso de, por exemplo, desenhos exemplificativos, gráficos, etc., com muito mais rapidez e perfeição, poupando inclusive tempo na preparação das aulas.

Não pretendendo ser exaustivo sobre o assunto — nem poderia, pois, uma vez mais, estaria a limitar a criatividade dos utentes —, focarei neste ponto, sobretudo a questão do *ensino assistido por computador*,

fazendo também uma breve referência à utilização eventual do equipamento numa aula normal. Todavia, este segundo aspecto será desenvolvido sobretudo no capítulo seguinte.

Segundo a definição do «Dictionnaire de l'informatique», já citado, o E.A.C. (Ensino Assistido por Computador) é «um conjunto de técnicas e métodos de utilização de sistemas informáticos como utensílios pedagógicos integrados no contexto educativo, seja no ensino inicial (primário, secundário ou superior), seja na formação permanente (profissional)». Tal definição implica, pois, um trabalho vasto, sectorizado e moroso, que terá que resultar, necessariamente, de uma *opção*. Ou seja, não é pensável que uma escola se possa decidir por este tipo de ensino, uma vez que tal implica não só uma equipa de vários especialistas (analista, programador, etc.) em informática, como obriga a uma reestruturação dos programas escolares, meios de avaliação, etc.

E se é verdade que tal opção altera favoravelmente as relações de trabalho professor/aluno, aluno/aluno, aluno/professor; individualiza a aprendizagem, acelerando-a e permitindo resultados imediatos (e a prazo) mais seguros e profundos; liberta o professor para tarefas mais criativas, permitindo-lhe avaliar mais correctamente o trabalho dos alunos. Se tudo isto (e outras coisas) é verdade, dizia, não o é menos que a opção por tal meio só é concebível se assumida a nível nacional ou, quando muito, regional e em experiência piloto, de qualquer modo com o assentimento do Ministério.

O professor torna-se *responsável* fundamentalmente pelo *objectivo de trabalho dos alunos*, intervindo sobretudo em função das necessidades destes e participando na aquisição das informações indispensáveis ao melhoramento dos programas utilizados, em conjunto com a equipa realizadora desses mesmos programas. Tem, sobretudo, uma tarefa de mediador entre o software utilizado e a eficácia do trabalho dos alunos, procurando adaptar as suas intervenções às necessidades pontuais surgidas no decorrer das sessões de trabalho. Fora das aulas, tem um importante papel de reflexão crítica, levando aos projectistas e programadores as conclusões necessárias a alterações que eventualmente se imponham.

O aluno passa a ter possibilidades de receber mensagens emitidas pelo sistema, compondo as suas próprias mensagens e, através destas trocas, estabelecer diálogo, cujo fulcro se deslocará, alternadamente, entre um e outro. Portanto, o aluno pode ser *solicitado* a responder — e essa resposta pode ser apenas um elemento para o desenvolvimento de um processo mais elaborado — assim como ter a liberdade de comandar o sistema, quer no sentido de o levar a executar determinadas tarefas, quer como consulta decorrente de matéria desconhecida (ou mal conhecida), verificação dessa decorrente de solicitação anterior. Durante a utilização, o aluno pode responder com o seu ritmo individual, tendo ainda possibilidade de utilizar funções pedagógicas anexas e diversificadas (ainda de uma forma individual).

Muito mais poderia dizer-se sobre o assunto. Todavia, a intervenção é tão só dar a conhecer uma possibilidade que, como atrás disse, não está ao alcance particular — segundo penso — de cada escola. O que

não impede o aproveitamento no E.A.G., mesmo para projectos menos ambiciosos mas possíveis de desenvolver.

No que respeita à utilização eventual do computador como auxiliar pontual em aulas «normais», o assunto será tratado, como disse, desenvolvendo-se no capítulo seguinte.

2.4 — Tempos livres e clube de informática

A escola é, genericamente, tomada (e apresentada) como uma instituição séria e com uma inerente gravidade ancestral. Alguns estabelecimentos decidiram, feliz e inteligentemente, libertar-se dessa austeridade deslocada e cada vez mais inaceitável, procurando ir ao encontro dos interesses daqueles — nunca é demais afirmá-lo — que são, na essência, a razão de existir da Escola: os alunos.

Sem pretender entrar aqui em profundas locubrações pedagógicas, dificilmente poderá pôr-se em dúvida a razão directa em que o insucesso escolar se encontra, relativamente ao maior afastamento que a Escola tem dos interesses da juventude. Penso mesmo que uma das características essenciais a uma *boa* escola é a capacidade de assimilar esses interesses, modificando-os num sentido pedagógico ou mesmo, se se verificar possível, utilizando-os sem transformação em estratégias adequadas.

De nada adianta — a experiência mostra-o, permanentemente — tentar contrariar qualquer evolução histórica, sobretudo quando o seu motor é o futuro: os jovens. Resta à geração anterior a esse futuro, aderir por forma a adquirir a confiança dos seus sujeitos, emprestando-lhes alguma experiência útil, sobretudo no sentido de evitar quedas no vazio.

O grande interesse inicialmente demonstrado pelos computadores, por parte dos jovens, ligou-se — e liga-se ainda — aos tempos livres. As possibilidades lúdicas, antes não sonhadas, que tais máquinas oferecem, de uma maneira fácil e que vai acompanhando, a par e passo, a imaginação mais fantástica, foram-nas naturalmente tornando no parceiro ideal dos jogos, sobretudo pelas totais possibilidades de simulação que possuem. A autêntica febre dos jogos de computador — que tornou a venda do software no já chamado «negócio do século» — faz correr a juventude muito mais rápida e esclarecidamente para um futuro que cada dia é mais possível, ainda que de forma simulada.

E se a nossa geração ficou marcada pelo jogo como forma de *vício* maléfico, destruidor de fortunas e famílias, responsável por crimes e, a par disso, fonte inesgotável de literatura do mais diverso género, algo de estranhamente contrário e positivo parece estar a suceder com os jogos de computador: começa a verificar-se — numa prática nem sequer forçada ou sugerida — constituírem uma firme e decisiva alternativa a outras utilizações destrutivas do tempo livre (e outro) dos jovens. Se pensarmos só um pouco, com tranquilidade e sem receios gratuitos, no que se passa em nossas casas com os nossos filhos, verificamos que: se é verdade que passam horas a «brincar» com o computador, essas são aquelas horas em que não sabíamos com quem estavam, onde estavam, nem a fazer o quê; o dinheiro da mesada que lhes damos — anteriormente utilizado

em tabaco ou droga — é agora ciosamente guardado para comprar o último jogo para o «Spectrum» ou um «joystick»; o próprio convívio entre os nossos filhos, tornou-se saudável e, acima de tudo, tranquilo e *pacífico*.

Estas são algumas verdades que reconhecerão os pais cujos filhos possuem um microcomputador, ainda que talvez «nunca tivessem pensado ou reparado nisso». Mas nem sequer são as únicas. Para além dos aspectos puramente sociais focados no parágrafo anterior, acresce a verificação de um aumento de agilidade mental por parte desses jovens, a par de outros factores físicos e psicológicos mais desenvolvidos; e ainda, com certa frequência, a passagem progressiva do simples interesse, manifestado inicialmente, pelos jogos, para o *trabalho* «sério» de programação, mesmo que numa fase intermédia tenha como objectivo inventar novos jogos.

Tudo isto cria, de qualquer modo e entre outras coisas, uma preparação diferente para a continuação dos estudos; desenvolve o espírito de investigação, disciplinando o raciocínio lógico; permite adquirir, bem mais cedo, certo grau de responsabilidade; etc.

A Escola — todas as escolas, se bem que em graus diferentes — possuem, de há muito, o cada vez mais grave problema dos tempos livres. Não há espaços para utilização controlada e *útil* desses tempos, pelos alunos; não haveria, sequer, pessoal suficiente para fiscalizar tal utilização; cada vez menos se divisa solução viável e continuada para o problema, enfim.

Do mesmo modo que fora da escola, o computador poderá contribuir para solucionar, mesmo que parcialmente, a questão.

Como, aliás, alguns estabelecimentos de ensino fizeram — para não falar já de departamentos estatais, a nível nacional — é perfeitamente possível organizar clubes de informática nas escolas. Não há receita única, claro, para a formação e desenvolvimento de tais clubes. Todavia — e aproveitando experiências já realizadas, quer em Portugal quer no estrangeiro — um bom método é fundar esses clubes a partir de um núcleo de alunos e professores interessados; criar os respectivos — e necessariamente simples — estatutos (regras, no fundamental); estabelecer uma jóia e quotização, a fim de criar fundos para aquisição de livros, revistas e software (e mesmo computadores e outro equipamento); e planificar a utilização do sistema, podendo assim a sala de informática, por exemplo, ter plena utilização.

É bem mais fácil do que possa parecer, e com resultados sempre muito positivos, em vários aspectos. Não só, como atrás disse, o problema dos tempos livres fica parcialmente solucionado, como a própria escola adquire uma dinâmica de funcionamento muito maior, no âmbito da informática, equipando-se muito rapidamente com hardware e software que, de outro modo, levariam anos a ser adquiridos. Para além do que nascerá e se desenvolverá, no seu interior, um sector de investigação muitíssimo importante e valioso, já que, entre outras coisas, levará a uma transformação positiva no relacionamento entre os vários elementos da escola, revalorizando as pessoas e permitindo trocas de experiências — e consequente aprendizagem — permanentes.

Vem a talhe de foice contar o que sucedeu com um colega da escola em que estou colocado, pessoa tímida e introvertida, com uma tão exagerada dificuldade de comunicação que nos levava, por vezes, a questionar como poderia dar aulas — actividade que exige, precisamente, grande capacidade de comunicação. Pois um dia, estando eu e outro colega, na sala de professores, a falar sobre a dificuldade de utilização de um simulador de voo para o «Spectrum», o Costa — nome do primeiro a que me referi — começou de súbito a falar connosco, muito entusiasmado:

— Vocês já conseguiram aterrar alguma vez?!
 — Eu consegui umas três vezes, apenas — respondi.
 — Eh, pá! Pois eu — prosseguiu — cada vez que vou tentar uma aterragem, e apesar de ter o cuidado de manter velocidade e altitude relacionadas, etc..., estampo-me.

É isto foi o início de um diálogo — nunca antes existente — que nos levou depois à discussão frequente de problemas mais sérios, de carácter pedagógico e científico, e não já apenas no domínio da informática.

Uma nota final, pois, que me parece importante, apesar de tudo o que até aqui escrevi: o vocábulo «club», bem como o termo «jogo», apenas marcaram, insisto, negativamente e por vezes de uma forma dramática, a geração anterior (sobretudo a nossa). É fundamental que essa geração se liberte de fantasmas que só a si pertencem, procurando entender abertamente o novo e positivo valor que tais palavras podem ter para os jovens de hoje.

Um breve exemplo

Apesar do que acabo de dizer acerca dos tempos livres, é evidentemente possível restringir a utilização do computador durante os mesmos — como o fez, de resto, o núcleo de informática da Escola da Bela Vista, que constitui o exemplo que passamos a apresentar.

Por iniciativa de um grupo de professores e alunos — e com o acordo do Conselho Directivo — fundou-se o referido núcleo em bases um pouco diferentes das da Escola de S. Julião.

Esse núcleo começou por estabelecer normas para associação ao «Clube de Informática»:

NORMAS PARA SE SER SÓCIO

1 — Qualquer aluno ou professor da Escola Secundária da Bela Vista pode ser sócio.

2 — Todos os sócios têm os mesmos direitos e deveres.

3 — Condições para a admissão de sócios:

a) que o sócio possua os conhecimentos básicos para a utilização e manuseio do computador

(LOAD “ “
 (SAVE “ “
 (ENTER
 (RUN
 (Ligações eléctricas
 (Ligação à televisão, gravador e computador)
 b) Caso o sócio não possua estes conhecimentos, eles ser-lhe-ão ministrados no curso de programação.
 c) o sócio que já saiba utilizar sem perigo e correctamente o computador receberá o seu cartão de sócio, que lhe permitirá requisitá-lo. Do cartão de sócio constará:
 — o nome do aluno/professor
 — o número, o ano e a turma
 — o número do seu Bilhete de Identidade
 — a sua fotografia
 — o carimbo da Escola Secundária da Bela Vista
 d) Todos os sócios deverão pagar mensalmente uma quota de 50\$00.
 e) Como o número de pessoas a frequentar o curso tem de ser limitado para que este seja profícuo, haverá uma lista de espera. As pessoas que estejam nessa lista de espera não necessitam de pagar quotas.

Ao contrário do que sucede na Escola de S. Julião, aqui o clube tem receita por quotização, o que lhe permite manter um fundo destinado à reparação ou aquisição do equipamento.

Inicialmente — como pode ver-se pelas normas de utilização que se apresentam a seguir —, foram muito mais restritivos no tipo de trabalho a executar com o material, embora neste momento pensem em alargar a utilização a outros domínios:

UTILIZAÇÃO DO COMPUTADOR

- 1 — O computador só poderá ser utilizado pelos sócios do núcleo.
- 2 — Essa utilização só pode ser feita fora do horário de funcionamento da biblioteca como sala de aula.
- 3 — A sala deve estar disposta de forma a que quem a esteja a utilizar como biblioteca não seja incomodado por quem utiliza o computador e vice-versa.
- 4 — O número limite de pessoas por computador que estejam simultaneamente a utilizá-lo é de 2. No caso de este ser utilizado numa aula deve ser manipulado por uma pessoa apenas.
- 5 — A inscrição para a utilização do computador deve ser feita no respectivo quadro de inscrição, afixado na biblioteca.
- 6 — Tendo em conta que o computador não deve ser monopolizado, mas permanecer sempre ao serviço de todos, não se deverá ultrapassar as duas horas diárias de utilização.
- 7 — A utilização do computador deverá restringir-se à aprendizagem de programação e à sua aplicação pedagógica. Não são permitidos, pois, os jogos ditos comerciais.

8 — Os prejuízos decorrentes de uma má utilização propositada e outros actos de vandalismo serão integralmente pagos pelo culpado e ser-lhe-á retirado o direito a ser sócio.

9 — Os prejuízos normais que ocorram não intencionalmente serão avaliados e pagos, na medida do possível, pelos fundos do núcleo.

10 — Para requisitar o computador o sócio terá de apresentar o seu cartão.

Tal como consta das normas de adesão, cada membro possui um cartão com identificação e fotografia, bem como o impresso da quota colado no verso.

Neste momento, o clube possui cerca de 100 associados —entre professores e alunos—, tendo já adquirido dois televisores e três computadores, preparando-se para comprar os primeiros periféricos.

APLICAÇÕES PRÁTICAS

- 3.1 — A utilização em algumas disciplinas
- 3.2 — Algumas rotinas e programas úteis
- 3.3 — Programas comerciais

3.1 — A utilização em algumas disciplinas

Seria necessária uma outra obra, ou talvez até várias, para apresentar — e não exaustivamente — listagens de programas para todas as disciplinas (ciências) existentes no ensino Preparatório e Secundário. Bem gostaria de o fazer nesta, já que, como disse no preâmbulo, um dos objectivos fundamentais é precisamente obter a adesão dos professores a este novo método (auxiliar) de trabalho que constitui a informática. Só que tal não seria viável editorialmente, nem prático, convenhamos.

Assim, procurarei dar aqui alguns exemplos de aplicação do computador como auxiliar de algumas disciplinas, tentando sobretudo alargar tais exemplos a áreas afastadas da matemática — estranhamente considerada como a disciplina por excelência da informática. A este aspecto me referirei mais adiante.

Os programas aqui apresentados são da autoria de professores, ou transcritos (e adaptados) de revistas ou ainda de minha autoria. Na selecção que fiz, procurei privilegiar programas e rotinas pouco extensos, sobretudo atendendo a que numa primeira fase será difícil, a quem se inicia, o trabalho de longas horas de «teclagem» de um programa. Apesar disso, alguns (poucos) são extensos — factor, todavia, claramente compensado pelos resultados que produzem.

Durante a elaboração deste capítulo do livro várias vezes me perguntei qual a divisão melhor, do ponto de vista do leitor — já que muitos dos programas ou rotinas têm aplicabilidade em várias disciplinas. Apesar de continuar com uma certa indecisão acerca do assunto (e de nem sequer estar convencido de ser a melhor a solução que constituiu a minha opção final), decidi fazer uma divisão destes por disciplinas do ensino.

De qualquer modo, é desejável (muito aconselhável, diria) que os professores, alunos e outros se não limitem à leitura das secções cujas disciplinas lhes interessam mais directamente, uma vez que haverá, estou certo, em todas elas rotinas ou blocos aplicáveis a outros programas.

Para evitar as longas listagens de três dos programas apresentados, a meio do texto — a fim de não obrigar os leitores a terem de saltar demasiadas folhas no momento da leitura simples —, decidi transferi-las para um apêndice especificamente dedicado às mesmas.

História

Um exemplo do que acabo de escrever é o programa «A Expansão Portuguesa no Séc. XV», cuja listagem se encontra no apêndice já referido. Concebido e elaborado por um professor de História — Carlos Catalão Alves — e aplicado em algumas das suas aulas com invariável sucesso, tal programa poderá servir perfeitamente — com pequenas alterações de texto — para a disciplina de *Português*, em núcleos programáticos relativos a Camões ou, mais especificamente, aos Lusíadas.

O programa compõe-se de duas partes:

1 — Em BASIC, toda a estrutura do programa, bem como o texto e ainda algumas rotinas;

2 — Em linguagem máquina, todos os ecrãs necessários à animação do texto. Esta parte do programa foi feita com o programa comercial «Paint Box», que é dos melhores existentes no mercado para o efeito — definição de gráficos e animação de imagens.

Segue-se uma sequência resumida das imagens e texto que vão surgindo no ecrã do televisor, à medida que o programa vai correndo. Terminado o texto correspondente a cada ecrã, há sempre uma opção (na parte de baixo do televisor) que permite o acesso a outra imagem ou a mais texto.



Fig. 21

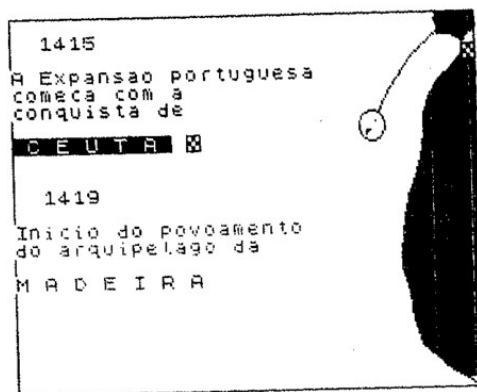


Fig. 22

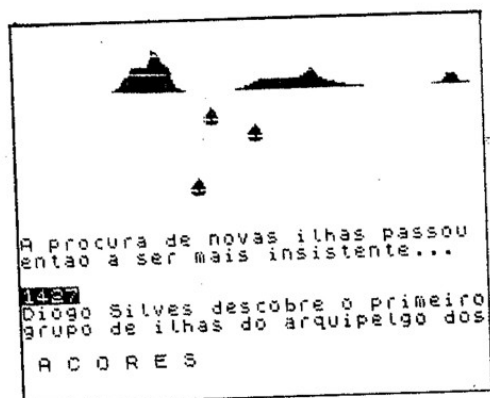


Fig. 23

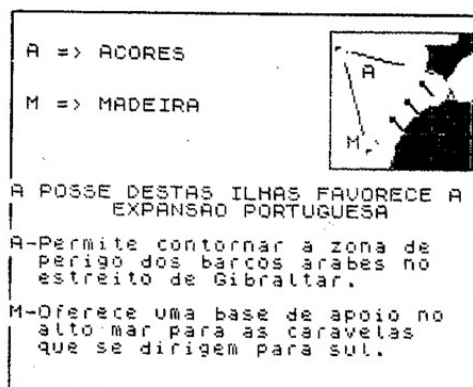


Fig. 24

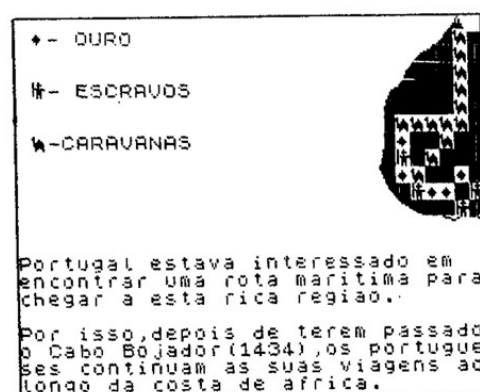


Fig. 26

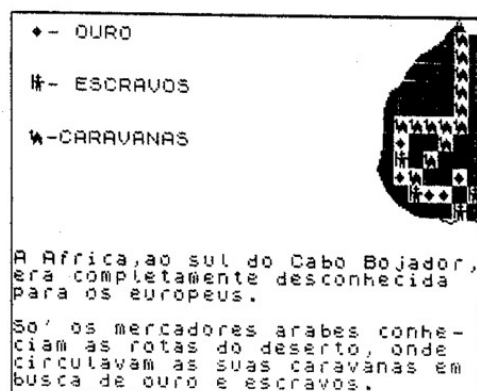


Fig. 25

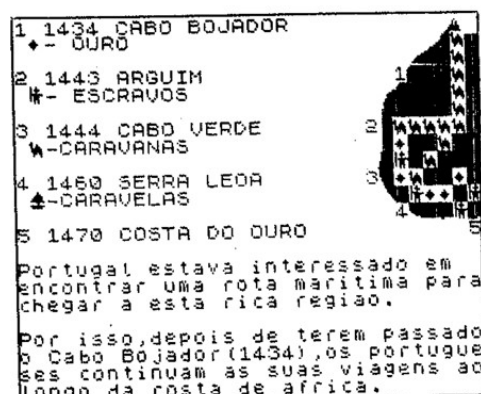


Fig. 26-A

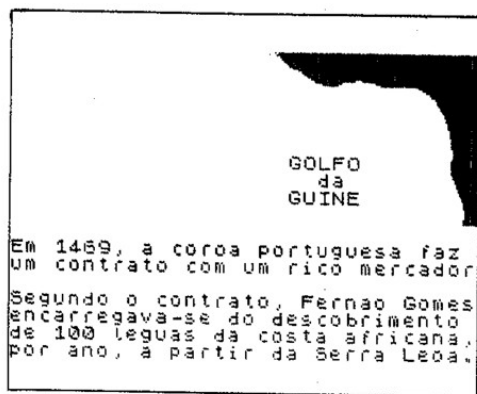


Fig. 26-B

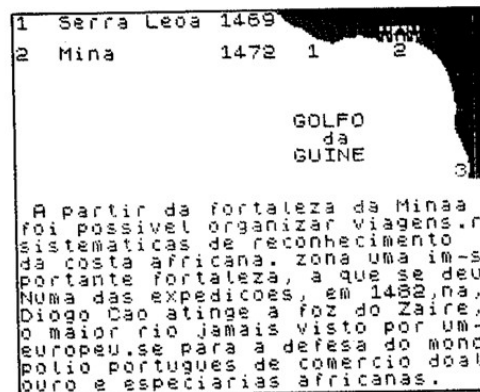


Fig. 27-A

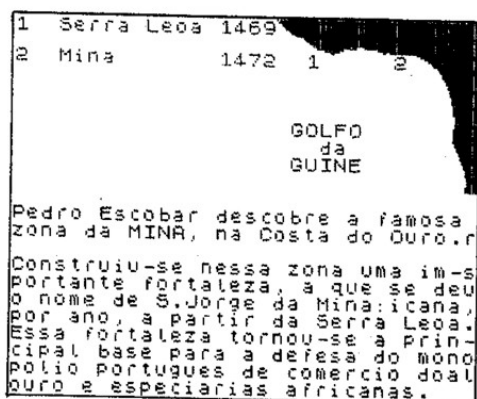


Fig. 27

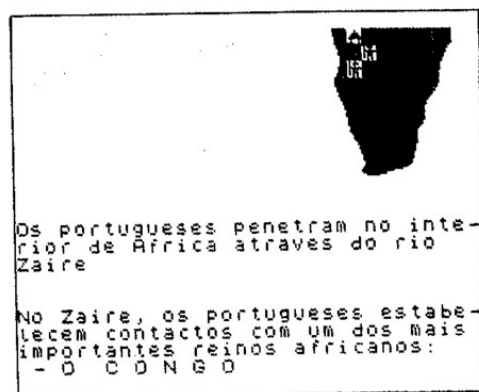


Fig. 28

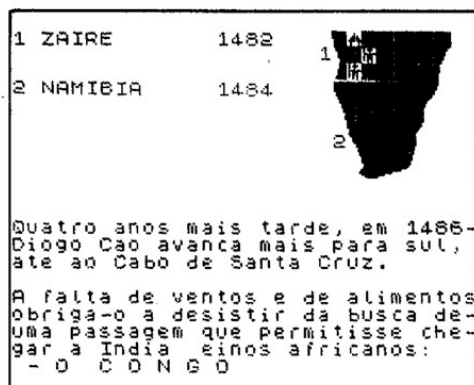


Fig. 28-A

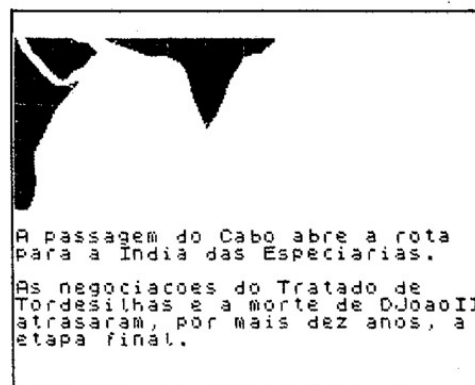


Fig. 29

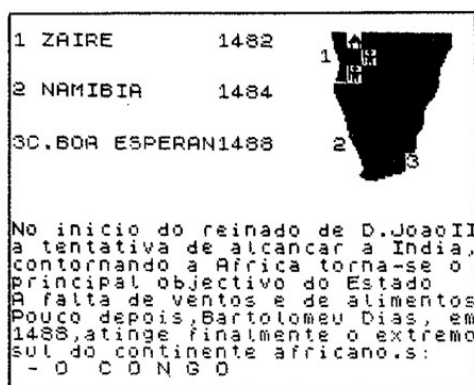


Fig. 28-B

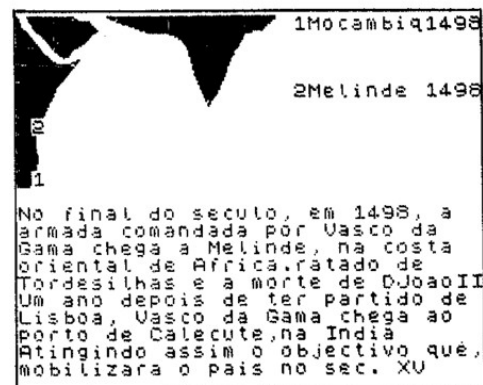


Fig. 29-A

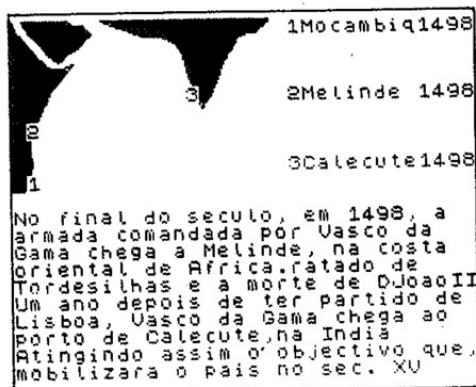


Fig. 29-B

Testes

Em princípio, os testes elaborados para computador — no domínio do ensino — não servem exactamente para proceder a uma avaliação substitutiva dos testes normais. A sua maior utilidade é, sobretudo, a de fornecer ao aluno a possibilidade de auto-controlo na aprendizagem, fornecendo igualmente indicações ao professor sobre as áreas da matéria acerca das quais persistem dúvidas.

Assim, o professor que elaborou o programa de História que acabámos de ver programou um teste sobre a Expansão Portuguesa no Séc. XV. As questões são de vários graus de dificuldade, só passando ao grau seguinte quando se verifica que o anterior está adquirido.

Exemplo de uma questão do nível 1:

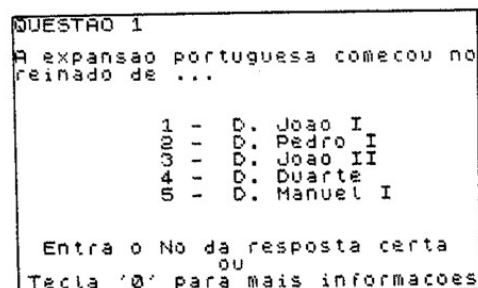


Fig. 29-C

E o exemplo de uma questão do nível 3:

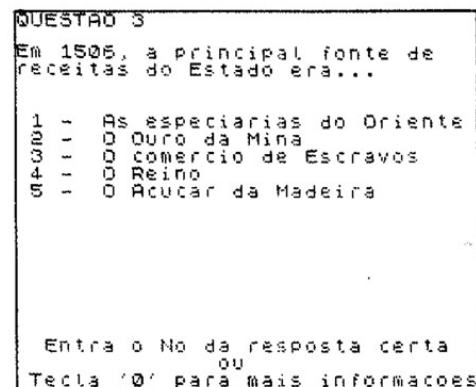


Fig. 29-D

Face à resposta do aluno, das duas uma: ou ela está correcta e o computador passa à seguinte; ou está errada e, então, consoante a pergunta, surge uma mensagem deste tipo, no écran:

ERRADO

consulte-se o mapa e a cronologia

As indicações de consulta são muito importantes, já que, para além dessa consulta propriamente dita, orientam o aluno no método de estudo, constituindo assim, também, um outro nível de aprendizagem.

Após o «menu»:

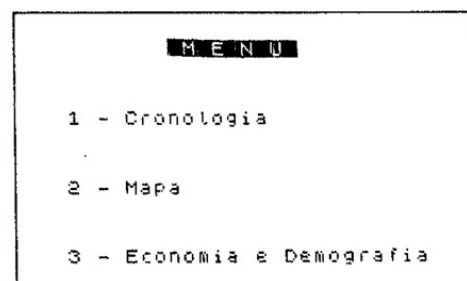


Fig. 30

E consoante a opção do aluno, surgirá um dos três écrans que podem ver-se em seguida, os quais, não fornecendo resposta tipo solução de livro de exercícios, dão toda a informação necessária para, obrigando de qualquer modo o aluno a um raciocínio, o levar a uma solução trabalhada.

C R O N O L O G I A			
1 - Reis de Portugal			
AFONSO H.	1120-1185	FILIPE I	1580-1598
SANCHO I	1185-1211	FILIPE II	1598-1621
AFONSO II	1211-1223	FILIPE III	1621-1640
SANCHO II	1223-1248	JOAO IV	1640-1656
AFONSO III	1248-1279	AFONSO VI	1656-1683
DINIS	1279-1325	PELRO II	1683-1706
AFONSO IV	1325-1357	JOAO V	1706-1750
PELRO I	1357-1367	JOSE	1750-1777
FERNANDO	1367-1385	MARIA I	1777-1816
JOAO I	1385-1433	JOAO VI	1816-1826
DUARTE	1433-1438	PELRO IV	1826-1828
AFONSO V	1438-1481	MIGUEL	1828-1834
JOAO II	1481-1495	MARIA II	1834-1853
MANUEL I	1495-1521	PELRO V	1853-1861
JOAO III	1521-1557	LUIS	1861-1889
SEBASTIAO	1557-1578	CARLOS	1889-1908
HENRIQUE	1578-1580	MANUEL II	1908-1910

Fig. 31

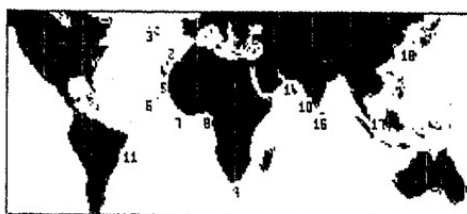
	
1- CEUTA.....1415	10- CALECUT.....1498
2- MADEIRA.....1415	11- BRASIL.....1500
3- ACORES.....1427	12- COCHIN.....1502
4- BORDOIR.....1434	13- CAMANOR.....1502
5- ARGUIN.....1443	14- DRNUZ.....1508
6- CARO VERDE.....1444	15- DOA.....1510
7- SERRA LEOA.....1460	16- CEILAO.....1510
8- MIMA.....1472	17- MALACA.....1513
9- C. DA ESPERANCA.....1472	18- MACAU.....1557

Fig. 32

RECEITA DO ESTADO			1506	1518
ANOS	REINO	ALFONSO DE LISBOA	REINO	ALFONSO DE LISBOA
1506	173.000	24.000	173.000	245.000
1518	245.000	40.000	245.000	40.000
1534	388.000	68.000	388.000	120.000
1557	607.000	101.000	607.000	11.000
1588	934.000	210.000	934.000	50.000
			ACORES	2.000
			ESPECIARIAS	17.000
				135.000
				300.000

POPULACAO			
REINO	LISBOA	PORTO	
1527	1.100.000	1527	65.000
1580	1.080.000	1584	100.000
1636	1.100.000	1590	120.000

POPULACAO / BRASIL			
ANOS	BRASIL	ESCRANOS	TOTAL
1546	2.000	4.500	6.500
1583	24.750	32.500	57.250
1600	30.000	120.000	150.000

Fig. 33

A listagem deste programa pode ser igualmente encontrada no apêndice já referido.

Educação visual (e Geometria)

O programa que se segue é bastante rudimentar. Todavia, tem o mérito de servir para o TS 1500, aproveitando bastante bem as capacidades do mesmo no domínio da imagem.

Trata-se, pois, de um programa de gráficos, possibilitando o desenho de círculos, elipses e outras figuras geométricas, quer no écran, quer, posteriormente, na impressora.

Antes de introduzir/carregar o programa é necessária a instrução POKE 16389, 121.

O programa tem uma curta rotina em código máquina, na linha 7000, a qual é carregada na instrução REM da linha 1, quando se faz RUN. Por isso, são absolutamente indispensáveis 38 espaços na instrução REM dessa linha (REM ... 38 espaços ...).

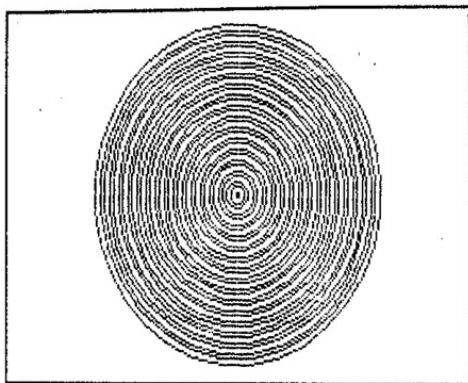


Fig. 34

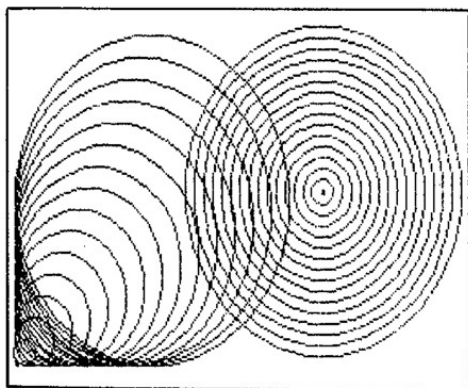


Fig. 35

```

1  DEM grafico TS1500
2  GO SUB 7000
3  RANDOMIZE USR 16516
4  GO SUB 5000
5  GO SUB 6000
6  LET M$=INKEY$
7  IF M$<>"K" AND M$<>"L" AND
8  M$<>"E" AND M$<>"C" THEN GO TO 6
9  GO SUB (11 AND M$="L")+ (850
0  AND M$="K")+ (3000 AND M$="C")+
(8000 AND M$="E")

```

```

9  LET A$=""
10 GO TO 5
11 CLS
12 PRINT AT 0,0;"DESENHO DE LI
13 NH"
14 PRINT AT 8,0;"VALOR DAS COO
15 RDENADAS X e Y, PARA O INICIO
16 DA LINHA";AT 15,5;"INIC.";AT 17
17 ,5;"X=";AT 19,6;"Y="
18 30 INPUT A
19 40 PRINT AT 17,6;A
20 50 INPUT B
21 60 PRINT AT 19,6;B
22 70 PRINT AT 8,0;"AGORA AS COOR
23 DENADAS X e Y DO FIM DA LIN
24 HA"
25 80 PRINT AT 15,20;"FIM";AT 17,
26 21;"X=";AT 19,21;"Y="
27 90 INPUT C
28 100 PRINT AT 17,23;C
29 110 INPUT D
30 120 PRINT AT 19,23;D
31 125 IF A<0 OR A>40 OR B<0 OR B
32 >40 OR C<0 OR C>40 OR D<0
33 THEN GO TO 11
34 130 PRINT AT 21,0;"PRESS. UMA T
35 ECLA P/TRACAR LINHA"
36 140 IF INKEY$<>" " THEN GO TO 14
37 0
38 150 IF INKEY$="" THEN GO TO 150
39 160 GO SUB 1000
40 170 GO TO 5
41 1001 RANDOMIZE USR 16534
42 1002 LET U=C-A
43 1010 LET V=D-B
44 1020 LET D1X=SGN U
45 1030 LET D1Y=SGN V
46 1040 LET D2X=SGN U
47 1050 LET D2Y=SGN V
48 1060 LET M=ABS U
49 1070 LET N=ABS V
50 1080 IF M>N THEN GO TO 1130
51 1090 LET D2X=0
52 1100 LET D2Y=SGN V
53 1110 LET M=ABS U
54 1120 LET N=ABS V
55 1140 LET S=INT (M/2)
56 1150 FOR I=0 TO M
57 1160 PLOT A,B
58 1170 LET S=S+N
59 1180 IF S<M THEN GO TO 1230
60 1190 LET S=S-M
61 1200 LET A=A+D1X
62 1210 LET B=B+D1Y
63 1220 GO TO 1250
64 1230 LET A=A+D2X
65 1240 LET B=B+D2Y
66 1250 NEXT I
67 1260 RANDOMIZE USR 16516
68 1261 IF A$="Y" THEN RETURN

```

```

1255 GO SUB 2000
1260 RETURN
2000 POKE 16416,0
2010 PRINT AT 23,0;"PRESS.UMA TE
CLA P/CONTINUAR
2020 IF INKEY$<>" THEN GO TO 20
2030 IF INKEY$="" THEN GO TO 203
0
2040 PRINT AT 23,0;"
2050 POKE 16416,2
2060 RETURN
3000 CLS
3010 PRINT AT 8,0;"INTR. AS COOR
DENADAS X e Y DO CENTRO DO CIR
CULO"
3020 PRINT AT 12,5;"COORD.:"
3030 PRINT AT 14,7;"X=";AT 16,7;
"Y="
3040 INPUT X
3050 PRINT AT 14,9;X
3060 INPUT Y
3070 PRINT AT 16,9;Y
3075 IF X>60 OR X<0 OR Y>40 OR Y
<0 THEN GO TO 3000
3080 PRINT AT 8,0;"INTR. X e Y R
ADIANOS PARA O CIRCULO OU
ELIPSE"
3090 PRINT AT 12,22;"RADI.:"
3100 PRINT AT 14,24;"X=";AT 16,2
4;"Y="
3110 INPUT XR
3120 PRINT AT 14,26;XR
3125 INPUT YR
3130 PRINT AT 16,26;YR
3135 PRINT AT 8,0;"QUANTOS PONTO
S (10-500)?"
3136 INPUT P
3137 PRINT AT 0,0;"QUER UMA LINH
A DE CADA PONTO AO CENTRO (S/
N)?"
3138 INPUT A$
3139 IF A$<>"Y" AND A$<>"N" THEN
GO TO 3138
3140 PRINT AT 21,0;"PRESS.UMA TE
CLA PARA DESENHAR"
3150 IF INKEY$<>" THEN GO TO 31
50
3160 IF INKEY$="" THEN GO TO 316
0
3170 CLS
3180 RANDOMIZE USR 16534
3190 FOR K=0 TO P
3200 LET XX=K/(P/2)*PI
3210 LET D1X=X+XR*SIN XX
3220 LET D1Y=Y-YR*COS XX
3230 IF D1X>60 OR D1X<0 OR D1Y>4
0 OR D1Y<0 THEN GO TO 3250
3240 PLOT D1X,D1Y

```

```

3245 IF A$="Y" THEN GO SUB 4000
3250 NEXT K
3255 RANDOMIZE USR 16516
3260 POKE 16416,0
3270 PRINT AT 23,0;"PRESS.UMA TE
CLA P/CONTINUAR
3280 IF INKEY$<>" THEN GO TO 32
80
3290 IF INKEY$="" THEN GO TO 329
0
3300 POKE 16416,2
3320 GO TO 5
4000 LET A=D1X
4010 LET B=D1Y
4020 LET C=X
4030 LET D=Y
4040 GO SUB 1002
4050 RETURN
5000 POKE 16514,25
5010 POKE 16515,3
5020 PRINT AT 0,12;"MENU";AT 1,1
2;"
5030 PRINT AT 4,5;"C....CIRCULO
OU ELIPSE";AT 6,5;"L....LINHA";A
T 8,5;"E....APAGAR DESENHO";AT 1
0,5;"K....TECLAS CURSOR"
5040 PRINT AT 14,0;"ESCOLHA A SU
A OPCAO"
5050 PRINT AT 19,0;"O COMPUTADOR
GUARDARA SEMPRE O DESENHO EM
MEMORIA, A MENOS QUE,FAÇA -RU
N."
5060 RANDOMIZE USR 16516
5070 POKE 16514,0
5080 POKE 16515,0
5090 RETURN
6010 POKE 16514,25
6020 POKE 16515,3
6030 RANDOMIZE USR 16534
6040 POKE 16514,0
6050 POKE 16515,0
6060 RETURN
7000 PRINT AT 0,5;"INTRODUZINDO
CODIGO MAQUINA";AT 3,10;"AGUARDE
POR FAVOR"
7001 LET A$="2A0440ED4B824009EB2
A0C40011903EDB0C92A0440ED4B82400
9ED5B0C40011903EDB0C9"
7002 LET X=1
7005 FOR I=16516 TO 16551
7010 LET X$=A$(X TO X+1)
7020 POKE I,16+CODE X$+CODE X$(2
)-476
7025 LET X=X+2
7030 NEXT I
7040 CLS
7050 RETURN
8000 CLS
8010 RANDOMIZE USR 16516
8020 RETURN

```

```

8500 CLS
8510 PRINT AT 10,0;"INTRODUZA AS
COORDEN. DE X e Y PARA O PONTO
DE INICIO"
8520 PRINT AT 13,5;"X=";AT 15,5;
"Y="
8530 INPUT X
8540 PRINT AT 13,7:X
8550 INPUT Y
8560 PRINT AT 15,7:Y
8565 IF X>60 OR X<0 OR Y>40 OR Y
<0 THEN GO TO 8500
8570 PRINT AT 21,0;"PRESS.UMA TE
CLA PARA COMECAR";AT 19,3;"PRESS
.X PARA SAIR DA ROTINA"
8580 PRINT AT 0,0;"USE OS CURSOR
ES S S 7 8 PARA LINHAS DIREIT
AS, O A P L PARA DIAGONAIS"
8590 IF INKEY$<>"" THEN GO TO 85
90
8600 IF INKEY$="" THEN GO TO 860
0
8610 RANDOMIZE USR 16534
8620 PLOT X,Y
8630 IF INKEY$="X" THEN GO TO 87
50
8640 LET X=X+(INKEY$="8" AND X<6
0)-(INKEY$="5" AND X>0)
8650 LET Y=Y+(INKEY$="7" AND Y<4
0)-(INKEY$="6" AND Y>0)
8660 IF INKEY$="0" THEN LET X=X-
(X>0)
8670 IF INKEY$="0" THEN LET Y=Y+
(Y<40)
8680 IF INKEY$="A" THEN LET X=X-
(X>0)
8690 IF INKEY$="A" THEN LET Y=Y-
(Y>0)
8700 IF INKEY$="P" THEN LET X=X+
(X<60)
8710 IF INKEY$="P" THEN LET Y=Y+
(Y<40)
8720 IF INKEY$="L" THEN LET X=X+
(X<60)
8730 IF INKEY$="L" THEN LET Y=Y-
(Y>0)
8740 GO TO 8620
8750 RANDOMIZE USR 16516
8760 RETURN
9997 CLEAR
9998 SAVE "***"
9999 RUN

```

Sendo a capacidade gráfica uma das mais espetaculares aplicações dos computadores, é nesse âmbito que pode encontrar-se um sem número de programas muito curtos (a par de outros bem mais complexos), com efeitos muito bons. Assim sendo, é sobretudo a disciplina de Educação Visual que maior partido pode tirar desta capacidade.

Seria impossível listar mesmo um número reduzido de programas curtos que dessem conta de *todas* as capacidades do computador. De qualquer modo, apresenta-se a listagem de alguns fundamentais, que poderão mesmo servir — integrados em programas maiores — para trabalhos mais desenvolvidos.

O programa base que se apresenta contém a estrutura fundamental para na mesma poderem ser implantadas as rotinas relativas às várias figuras, rotinas essas apresentadas posteriormente.

PROGRAMA 1

```

10 CLS
20 PRINT AT 0,10;"M E N U"
30 PRINT " 1 - QUADRADOS"
40 PRINT " 2 - POLIGONOS"
50 PRINT " 3 - POLIGONOS/VERT
ICES"
60 PRINT " 4 - CIRCULO"
70 PRINT " 5 - ESPIRAL"
80 PRINT " 6 - ENVELOPE"
90 PRINT " 7 - EUCLIDIANO"
92 PRINT " 8 - SPIRO"
94 INPUT A
96
98
100 REM programa principal/poli
9090
110 LET start=9700: LET setorig
in=9600: LET moveto=9500: LET li
neto=9400: LET polygon=300
120 LET HORIZ=30: LET VERT=20
130 GO SUB start
140 LET XMOVE=HORIZ*.5: LET YM
OVE=VERT*.5
150 GO SUB setorigin
159 REM input vertices poligono
160 DIM X(50): DIM Y(50)
170 INPUT "Numero de vertices "
;NPOL
180 FOR I=1 TO NPOL
190 INPUT ("X("+STR$ I+")");X(I)
, ("Y("+STR$ I+")");Y(I)
200 NEXT I
210 GO SUB polygon
220 STOP
230

```

```

300 REM poligono
310 LET XPT=X(NPOL): LET YPT=Y(
NPOL): GO SUB moveto
319 REM juncao dos vertices
320 FOR I=1 TO NPOL
330 LET XPT=X(I): LET YPT=Y(I):
GO SUB lineto
340 NEXT I
350 RETURN
360 DRAW XPIX,0,SGN N*PI
370 LET S=-S
380 NEXT I
390 RETURN
400 REM QUADRADO
410 LET start=9700: LET setorig
in=9500: LET moveto=9500: LET li
neto=9400
420 LET HORIZ=30: LET VERT=20
430 GO SUB start
440 LET XMOVE=HORIZ*0.5: LET YM
QUE=VERT*0.5
450 GO SUB setorigin
460 LET XPT=7.5: LET YPT=7.5: G
O SUB moveto
470 LET XPT=-7.5: LET YPT=7.5:
GO SUB lineto
480 LET XPT=-7.5: LET YPT=-7.5:
GO SUB lineto
490 LET XPT=7.5: LET YPT=-7.5:
GO SUB lineto
492 LET XPT=7.5: LET YPT=7.5: G
O SUB lineto
500 REM VERTICES
510 LET start=9700: LET setorig
in=9500: LET moveto=9500: LET li
neto=9400
520 LET HORIZ=3: LET VERT=2.1
530 GO SUB start
540 LET XMOVE=HORIZ*0.5: LET YM
QUE=VERT*0.5
550 GO SUB setorigin
560 DIM X(30): DIM Y(30)
570 INPUT "Valor de N ";N
572 LET ALPHA=0: LET ADIF=2*PI/
N
574 FOR I=1 TO N
576 LET X(I)=COS ALPHA: LET Y(I
)=SIN ALPHA
578 LET ALPHA=ALPHA+ADIF
580 NEXT I
582 FOR I=1 TO N-1
584 FOR J=I+1 TO N
586 LET XPT=X(I): LET YPT=Y(I):
GO SUB moveto
588 LET XPT=X(J): LET YPT=Y(J):
GO SUB lineto
590 NEXT J
592 NEXT I
595 STOP
600 REM CIRCULO

```

```

610
620
630
640
650
660
670
680
690
700 REM ESPIRAL
710
720
730
740
750
760
770
780
790
800 REM ESPIRAL CELTICA
810
820
830
840
850
860
870
880
890
900 REM ENVELOPE
910
920
930
940
950
960
970
980
990
1000 REM EUCLIDES
1010
1020
1030
1040
1050
1060
1070
1080
1090
1100 REM lineto
1110 REM OUT - XPEN,YPEN
1120 LET NXPEN=FN X(XPT)
1130 LET NYPEN=FN Y(YPT)
1140 PLOT XPEN,YPEN
1150 DRAW NXPEN-XPEN,NYPEN-YPEN
1160 LET XPEN=NXPEN: LET YPEN=N
YPEN
1170
1180 RETURN
1190 REM moveto
1200 REM IN - XPT,YPT

```

```

9500 REM OUT - XPEN,YPEN
9510 LET XPEN=FN X(XPT)
9520 LET YPEN=FN Y(YPT)
9530 RETURN
9540 REM set origin
9550 INPUT "Origem de x":XORIG
9560 INPUT "Origem de y":YORIG
9570 INPUT "Fim de x":XMOVE
9580 INPUT "Fim de y":YMOVE
9590 REM out - XORIG,YORIG,XPEN,YPEN
9600 LET XORIG=XORIG+XMOVE: LET YORIG=YORIG+YMOVE
9610 LET XPEN=FN X(XPT)
9620 LET YPEN=FN Y(YPT)
9630 RETURN
9640 DEF FN X(Z)=INT ((XORIG+Z)*XSCALE+0.5)
9650 DEF FN Y(Z)=INT ((YORIG+Z)*XSCALE+0.5)
9660 RETURN
9670 REM start
9680 INPUT "Horizontal"; HORIZ
9690 INPUT "Vertical"; VERT
9700 REM OUT - NXPIX,NYPIX,XORIG,YORIG,XYSCALE,XPEN,YPEN
9710 LET XORIG=0: LET YORIG=0
9720 LET XPEN=0: LET YPEN=0
9730 LET NXPIX=256: LET NYPIX=17
9740 LET XSCALE=NXPIX/HORIZ: LET YSCALE=NYPIX/VERT
9750 IF XSCALE>YSCALE THEN LET XSCALE=YSCALE
9760 RETURN
9770 REM plot/CalComp
9780 REM in - XPT,YPT,XPEN,YPEN,XORIG,YORIG,MODE
9790 INPUT "Ponto x":XPT
9800 INPUT "Ponto y":YPT
9810 INPUT "Traco x":XPEN
9820 INPUT "Traco y":YPEN
9830 INPUT "Origem x":XORIG
9840 INPUT "Origem y":YORIG
9850 INPUT "MODE":MODE
9860 REM out - XPEN,YPEN,XORIG,YORIG
9870 LET NXPEN=FN X(XPT)
9880 LET NYPEN=FN Y(YPT)
9890 IF ABS (MODE)=2 THEN PLOT XPEN,YPEN: DRAW NXPEN-XPEN,NYPEN-YPEN
9900 LET XPEN=NXPEN: LET YPEN=NYPEN
9910 IF MODE<0 THEN LET XORIG=XORIG+XPT: LET YORIG=YORIG+YPT
9920 RETURN

```

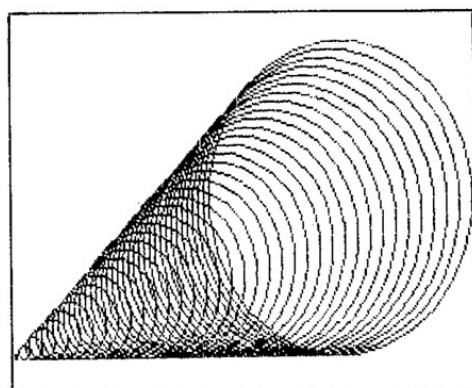


Fig. 36

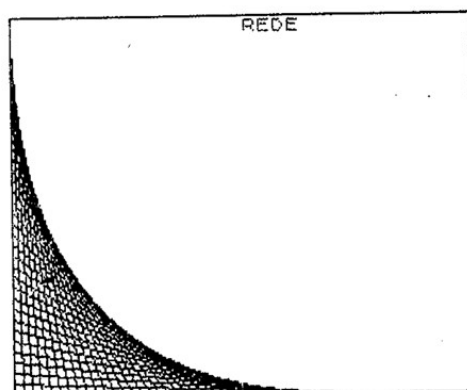


Fig. 37

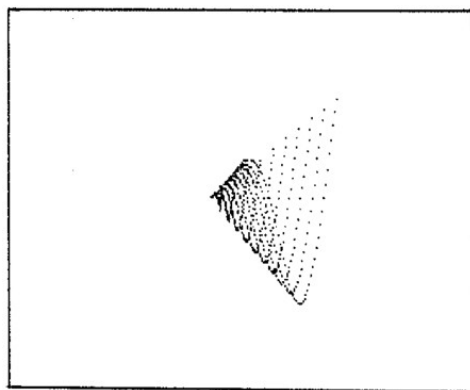


Fig. 38

As rotinas seguintes possibilitam a criação de matrizes de diverso tipo, podendo ser alterados os valores de X, Y e ANGLE, por forma a produzir efeitos diferentes:

ROTINA 1

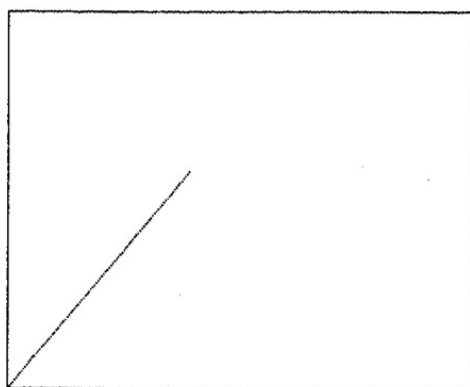


Fig. 39

ROTINA 2

```

10 OVER 1
20 LET lines=400
30 LET A=0: LET ANGLE=2*PI/lin
40 FOR I=1 TO lines
50 LET X=85*COS A
60 LET Y=85*SIN A
70 PLOT 128,88
80 DRAW X,Y
90 LET A=A+ANGLE
100 NEXT I
110 OVER 0
120 STOP

```

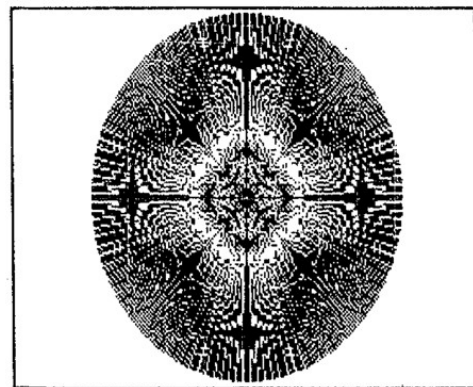


Fig. 40

ROTINA 3

```

10 OVER 1
20 LET x=INT (RND*255)
30 LET y=INT (RND*175)
40 PLOT x,y
50 BEEP 0.05, (x-y)/10
60 GO TO 20

```

ROTINA 4

```

10 OVER 1: PAPER 0: INK 7: BOR
DER 4: CLS
20 LET speed=2
30 LET x=0: LET y=0

```

```

40 LET xadd=speed: LET yadd=sp
eed
50 PLOT x,y
60 LET oldx=x: LET oldy=y
70 LET x=x+xadd
80 IF x>255-speed OR x<speed T
HEN LET xadd=-xadd
90 LET y=y+yadd
100 IF y>174-speed OR y<speed T
HEN LET yadd=-yadd
110 PLOT x,y
120 PLOT oldx,oldy
130 GO TO 50

```

Por fim, uma rotina muito curta e simples que, a partir de uma função definida na linha 50, traça figuras geométricas:

ROTINA 5

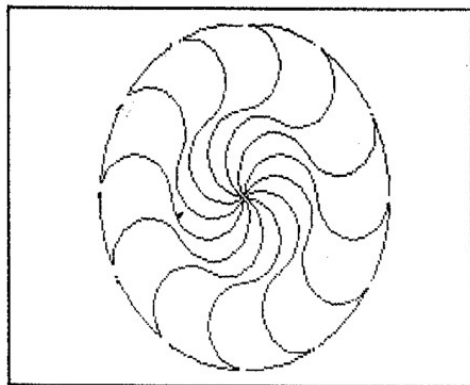


Fig. 41

```

10 CIRCLE 128,88,80: LET n=12
20 LET a=0: LET adif=2*PI/n
30 FOR i=1 TO n
40 PLOT 128,88
50 LET x=40*COS a: LET y=40*SI
N a
60 DRAW x,y,-PI: DRAW x,y,PI
70 LET a=a+adif
80 NEXT i

```

Alterando as funções, podem modificar-se, como é óbvio, os resultados.

Desenho de precisão

Apesar de existirem no mercado alguns programas destinados a desenho de precisão, penso ser vantajoso apresentar um programa relativamente simples mas muito poderoso.

De facto, este programa permite traçar linhas em 8 direcções, apagar os pontos que se entender, levantar o lápis para interromper um traçado, guardar um desenho em memória ou em cassette, apagar o écran, etc. As coordenadas de X e Y são indicadas na parte inferior do écran, o que possibilita uma exactidão absoluta nas medidas.

O programa permite, pois, efectuar trabalhos sérios como, por exemplo, projecções ortogonais, projectos rudimentares de edifícios, etc.

```

1 BORDER 2: INK 0: BRIGHT 1:
PAPER 5: CLEAR: CLS: BEEP .5,5
0
2 FOR x=10 TO 50: BEEP x/500,
x: NEXT x
3 PRINT FLASH 1: INK 2: PAPER
7: AT 10,12: "DESENHO": AT 13,2: "P
RIMA UMA TECLA PARA COMECAR"
4 BORDER 1: BORDER 1: BORDER
2: BORDER 2: PAUSE 1: BORDER 3:
IF INKEY$="" THEN GO TO 4
5 BRIGHT 0: CLS
6 PRINT "COMANDOS: " "1,2,3,4
=DIAGONAIS" "5,6,7,8=CURSORES"
"0=LEVANTAR LAPIS" "x=BORRACHA
COM 1,2,3,4,5,6,7 e 8" "i=BAIX
A LAPIS" "b=APAGA O DESENHO"
c=CIRCULO" "g=MEMORIA" "s=GRAU
A DESENHO" "p=MUDA CORES"
20 LET ni=0: LET no=0: LET bo=
0
25 PAUSE 0: CLS: PRINT "Para
os circulos, o raio deva ser menor
que as coordenadas do PLOT"
30 INPUT "Cor do PAPER e BORDE
R, 0-7 ?": p: "Cor da tinta, 0-7 ?"
t: BORDER p: PAPER p: INK t: CL
S
35 INPUT "Coord. X (0 a 255) ?"
a: "Coord. Y (0 a 175) ?": b
40 FOR c=1 TO 12288
41 IF INKEY$="1" THEN LET a=a-
1: LET b=b+1
42 IF INKEY$="2" THEN LET a=a-
1: LET b=b-1
43 IF INKEY$="3" THEN LET a=a+
1: LET b=b-1
44 IF INKEY$="4" THEN LET a=a+
1: LET b=b+1

```

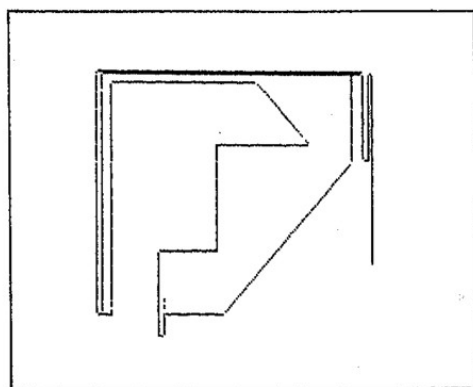


```

45 IF INKEY$="5" THEN LET a=a-
1 46 IF INKEY$="8" THEN LET a=a+
1 47 IF INKEY$="6" THEN LET b=b-
1 48 IF INKEY$="7" THEN LET b=b+
1 49 IF INKEY$="b" THEN CLS
50 IF INKEY$=" " THEN GO TO 50
51 IF INKEY$="s" THEN GO TO 99
9 52 IF INKEY$="0" THEN LET b0=1
53 IF INKEY$="i" THEN LET b0=0
54 IF INKEY$="x" THEN LET b0=2
55 IF a<0 THEN LET a=0
56 IF a>255 THEN LET a=255
57 IF b<0 THEN LET b=0
58 IF b>175 THEN LET b=175
59 IF INKEY$="c" THEN GO TO 40
0 65 IF INKEY$="g" THEN GO TO 20
0 66 IF b0=2 THEN PLOT a,b: BEEP
.001,50: PLOT OVER 1;a,b
67 IF b0=0 THEN PLOT a,b: POKE
30000+c,a: POKE 42389+c,b
68 PRINT #0;AT 0,0;"X=";a;" "
;#0;"Y=";b;" "
70 NEXT c
200 CLS: INPUT "Passos,1-5 ?";
e: FOR d=1 TO c-1 STEP e
210 PLOT PEEK (30000+d),PEEK (4
2259+d)
220 NEXT d
300 BEEP .1,50: PAUSE 0: GO TO
41 400 INPUT "RAIO ";r: IF r<a AND
r<b AND r>0 THEN CIRCLE a,b,r:
GO TO 58
401 GO TO 400
900 INPUT BRIGHT 1;"NOVA COR DO
PAPEL E DO BORDER ?";p;"COR DA
NOVA TINTA ?";t
901 PAPER p: INK t: FOR v=0 TO
21: PRINT OVER 1;" ": NEXT v: BOR
DER p: GO TO 63
999 INPUT "Nome do desenho ?";a
$: SAVE a$SCREEN$: GO TO 45

```

Fig. 41-A



Línguas

Já atrás se deu um exemplo da aplicabilidade de programas ao ensino do Português, com o programa da Expansão no Séc. XV. Todavia, é possível aplicar o computador ao ensino das línguas estrangeiras e com métodos muito elaborados.

O programa que se segue é muito simples e destina-se, sobretudo, a apresentar uma hipótese não desenvolvida — uma base de trabalho. Trata-se de um programa de vocabulário francês, o qual guarda perguntas e respostas em comandos DATA, podendo facilmente ser expandido e adaptado a outras necessidades.

Pode ser alterado em qualquer altura, nomeadamente no que respeita ao número de palavras em memória, bem como ao grau de dificuldade, devendo ser utilizado individualmente pelo aluno, que assim poderá testar os seus conhecimentos e aprender (o programa apresenta a resposta certa, caso haja erro). Por exemplo, poderá servir, adaptado, para o vocabulário de uma determinada unidade didáctica.

Na versão apresentada trata-se apenas de palavras, mas nada impede que nas linhas de DATA sejam colocadas frases (por exemplo com estruturas gramaticais de determinada unidade). Na verdade, trata-se de um bom programa-base para ser adaptado a uma série diversa de situações de ensino.

```

10 REM VOCABULARIO FRANCES
20 LET SCORE=0: PAPER 1: INK 7
: BORDER 1: CLS
30 DIM A(20)
40 PRINT "TAB 6; ISTO E UM TE
STE DE"

```

```

50 PRINT "TAB 8: VOCABULARIO
DE FRANCES"
60 PRINT "TAB 6: QUANTAS PALA
VRAS? (1 a 20)"
80 INPUT B
90 IF B<1 OR B>20 THEN GO TO 8
0
100 FOR C=1 TO B
110 FOR D=1 TO INT (RAND*20)+1
120 READ E$: READ F$: READ X
130 NEXT D
140 RESTORE : CLS
150 IF A(X)=1 THEN GO TO 110
160 LET A(X)=1
170 PRINT "PERGUNTA N.": C: "QUAL A TRADUCAO D
E": TAB 4: E$: "?"
190 INPUT A$
200 IF A$=F$ THEN LET SCORE=SCO
RE+1: BEEP .1,2,S*SCORE: PRINT F
LASH 1: INK 2: PAPER 5: "ESTA
CORRECTO"
210 IF A$<>F$ THEN PRINT "NAO
! EM FRANCES, "E$: "E": F$
220 IF C<B THEN PRINT "INK 3;
FLASH 1: BRIGHT 1: "A SUA MEDIA
E": SCORE: "
230 PAUSE 150
240 NEXT C
250 INK 2: PAPER 7: FLASH 1: CL
S: PRINT AT 5,3: "NESTE TESTE SO
MOU": SCORE
260 PAUSE 200: FLASH 0
270 RUN
280 DATA "NUNCA", "JAMAIS", 1, "SE
MPRE", "TOUJOURS", 2, "ONTEM", "HIER
", 3, "DEPOIS", "APRES", 4, "DESDE", "
DEPUIS", 5
290 DATA "MUITO", "TRES", 6, "MAIS
", "PLUS", 7, "MENOS", "MOINS", 8, "SI
M", "OUI", 9, "NAO", "NON", 10
300 DATA "OBRIGADO", "MERCI", 11,
"IR", "ALLER", 12, "SER", "ETRE", 13,
"TER", "AVOIR", 14, "BOM DIA", "BONJ
OUR", 15
310 DATA "PAI", "PERE", 16, "MAE",
"ME", 17, "PRIMO", "COUSIN", 18, "A
VO", "GRAND PERE", 19, "CASA", "MAIS
ON", 20
320 STOP

```

Um segundo programa/rotina (igualmente muito rudimentar) destina-se, basicamente, a análise de frases. Neste caso, contando o número de palavras que uma determinada frase contém.

Embora aparentemente a rotina não tenha grande serventia, aplicada a um programa mais elaborado poderá servir para estudos sintáticos:

```

100 REM  analise de frases
110 CLS
140 PRINT AT 10,4: INVERSE 1;"I
ntroduza a frase"
150 INPUT LINE S$
160 PRINT AT 10,0: TAB 31
210 LET CC=1: LET WC=0
220 LET WC=WC+1
230 LET CC=CC+1: IF S$(CC)<>" "
AND S$(CC)<>".," THEN GO TO 230
240 IF S$(CC)<>".," THEN GO TO 2
20
400 PRINT "O numero de P
alavras
WC: "

```

Geografia

O programa que se apresenta a seguir, sendo simples, possui tantas potencialidades que se torna difícil seriá-las. Tendo sido concebido para o traçado de mapas, em Geografia, pode ainda servir para Educação Visual e outras disciplinas.

Como pode ver-se é bastante curto, rápido de introduzir no computador e facilmente alterável, caso se pretenda melhorar. De qualquer modo — e assim mesmo como é apresentado — pode produzir linhas *ponto a ponto*, em oito direcções: verticais, horizontais e diagonais, conforme o esquema de teclas seguinte.



Após a introdução do programa pode seleccionar-se o ponto do écran em que pretende iniciar-se o desenho, e quando se terminar basta carregar na tecla "S".

Seguidamente, poder-se-á imprimi-lo com a impressora, bem como guardar o écran com a instrução

SAVE "nome do desenho" SCREENS.

Claro que este programa poderá servir, então, de base para algo de mais elaborado, com texto, por exemplo, quer para explicar matéria quer para teste.

```

10 BORDER 6
20 INPUT "Coordenada inicial d
e -X-"; X
30 INPUT "Coordenada inicial d
e -Y-"; Y

```

```

40 PLOT X,Y
70 LET C$=INKEY$
80 LET X=X+(C$="U")+(C$="J")+(
C$="M")-(C$="T")-(C$="G")-(C$="B
")
90 LET Y=Y+(C$="T")+(C$="Y")+(
C$="U")-(C$="B")-(C$="N")-(C$="M
")
100 LET X=X-(X>255)+(X<0)
110 LET Y=Y-(Y>175)+(Y<0)
120 IF C$="S" THEN STOP
130 GO TO 40

```

Segue-se um mapa fictício destinado apenas a mostrar as potencialidades do programa. Como pode ver-se, a definição do gráfico é muito boa, permitindo os mais ínfimos recortes — muito úteis para fronteiras naturais, rios, etc.

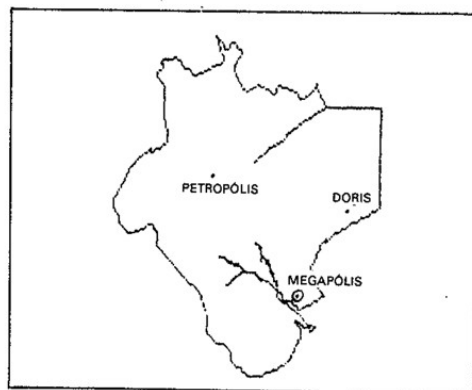


Fig. 42

As potencialidades do pequeno programa tornam-se ainda mais evidentes no desenho seguinte, que mostra a exactidão geométrica que o traço pode adquirir — se assim se desejar.

(E aqui está outro exemplo de como uma mesma rotina ou programa pode ter aplicabilidade em várias disciplinas.)

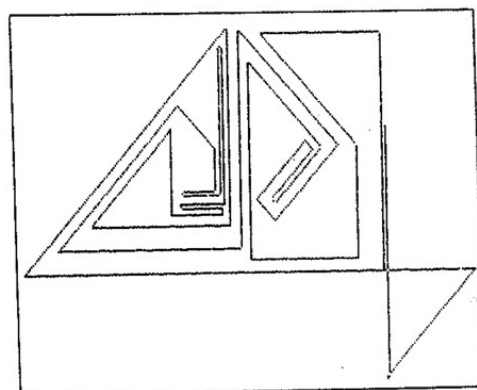


Fig. 43

Constelações

O programa que se segue destina-se ao TS 1500, devendo ser-lhe feitas as necessárias alterações caso pretenda introduzir-se no Spectrum ou no TC 2068. A terminologia encontra-se em Latim, podendo igualmente ser alterada para Português.

```

10 POKE 23658,8
20 FOR I=1 TO 3
30 FOR J=0 TO 14
40 GO SUB J*20+500
50 PRINT TAB 10;C$
60 GO SUB 1000
70 PAUSE 100/I
80 CLS
90 NEXT J
100 PRINT "QUER TENTAR O TESTE
? (S/N)"
110 INPUT X$
120 IF X$(1)="S" THEN GO TO 150
130 NEXT I
140 PRINT "EXPERIMENTE O TESTE
POR FAVOR"
150 PRINT " GRAU DE DIFICULDADE
1,2,3."
160 INPUT A
170 LET D=3-A
180 LET S=0
185 PRINT
190 PRINT "DIGITE CADA LETRA SE
PARADAMENTE SEGUIDA DE 'ENTER'"
191 PRINT

```



```

1040 NEXT L
1050 RETURN
1060 PRINT TAB VAL (a$(L TO L+1)
);
1070 LET L=L+1
1080 GO TO 1040
1090 PRINT
1100 GO TO 1040

```

Climas

E aqui está um programa destinado ao estudo dos climas, a partir da análise e classificação dos regimes termopluviométricos.

Após fazer RUN, podem introduzir-se os valores de precipitação dos 12 meses do ano. Feito isto, o computador traça os 12 gráficos de barras correspondentes (a azul, no Spectrum e TC 2068; a cinzento no TS 1500).

Em seguida podem introduzir-se as temperaturas, após o que são igualmente traçados os gráficos de barras correspondentes.

Terminada a operação, o computador classifica de imediato o tipo de clima, apresentando-o no écran, a par das médias, com um desenho do tipo do apresentado:

```

1 INK 0
5 DIM p(12)
10 REM Analise e classificacao
de regimes termopluviometricos
30 PRINT "Prec.(mm)"
40 PRINT
50 FOR n=1 TO 12
60 READ a$
70 PRINT a$
80 NEXT n
90 DATA "JAN","FEV","MAR","ABR",
"MAI","JUN","JUL","AGO","SET",
"OUT","NOV","DEZ"
100 LET c=1
110 FOR n=1 TO 12
120 INPUT p(n)
130 PRINT AT c+1,4;p(c)
140 LET c=c+1
150 NEXT n
155 LET c=1
160 FOR m=1 TO 12
170 FOR n=1 TO p(m)/10
172 INK 5
175 PLOT 64+c,8+n
176 PLOT 65+c,8+n
180 PLOT 66+c,8+n
190 NEXT n
195 LET c=c+16
200 NEXT m
205 INK 0
210 LET pt=0
220 FOR n=1 TO 12
230 LET pt=pt+p(n)
240 NEXT n
250 PRINT AT 15,0;"P.T.:";pt;"mm"

```

```

250 PRINT AT 0,11;"Temp.(c)"
270 DIM t(12)
280 FOR n=1 TO 12
290 INPUT t(n)
300 PRINT AT n+1,13;t(n)
310 NEXT n
315 LET c=1
316 INK 1
317 FOR m=1 TO 12
320 FOR n=1 TO t(m)/2
330 PLOT 72+c,8+n
332 PLOT 73+c,8+n
333 PLOT 74+c,8+n
340 NEXT n
350 LET c=c+16
350 NEXT m
370 INK 0
380 REM Calculo da temp. maxima
385 LET q=0
390 FOR n=1 TO 12
400 IF q<t(n) THEN LET q=t(n)
410 NEXT n
420 PRINT AT 16,0;"T.Max.:";q
430 REM Calculo da temp. minima
440 LET w=100
450 FOR n=1 TO 12
460 IF w>t(n) THEN LET w=t(n)
470 NEXT n
480 PRINT AT 17,0;"T.Min.:";w
490 LET x=q-w
495 PRINT AT 18,0;"A.T.:";x
500 REM Analise de regimes term
opluviometricos
510 LET d=0
520 IF w>18 AND x<3 AND pt>1500
THEN LET d=d+1
530 FOR n=1 TO 12
540 IF p(n)>60 THEN LET d=d+1
550 NEXT n
560 IF d=13 THEN PRINT AT 8,16;
"CLIMA EQUATORIAL": STOP
570 IF w>18 AND x<10 AND pt<1000
AND p(1)<10 AND p(2)<10 AND p(3)
<10 AND p(11)<10 AND p(12)<10
THEN PRINT AT 8,17;"CLIMA TROPIC
AL": "HUMIDO": STOP
580 IF w>18 AND x<10 AND pt<800
AND p(1)<10 AND p(2)<10 AND p(3)
<10 AND p(4)<10 AND p(10)<10 AN
D p(11)<10 AND p(12)<10 THEN PRI
NT AT 8,17;"CLIMA TROPICAL": "SE
CO": STOP
590 IF w>0 AND x<16 AND pt<150
THEN PRINT AT 8,16;"CLIMA DESERT
ICO": STOP
600 IF w>0 AND q>22 AND pt<500
AND p(5)<30 AND p(6)<30 AND p(7)
<30 AND p(8)<30 AND p(9)<30 THE
N PRINT AT 8,qv;"CLIMA SUBTROPIC
AL": "SECO (MEDITERRANEO)": STOP

```

```

510 IF w>0 AND q>=22 AND pt>=75
0 THEN PRINT AT 8,16;"CLIMA SUBT
ROPICAL";"HUMIDO": STOP
520 IF w>0 AND q<22 AND pt>750
THEN PRINT AT 8,16;"CLIMA TEMPER
ADO";"MARITIMO": STOP
530 IF q>=22 AND w<0 AND pt<750
THEN PRINT AT 8,16;"CLIMA TEMPE
RADO";"CONTINENTAL"
540 STOP

```

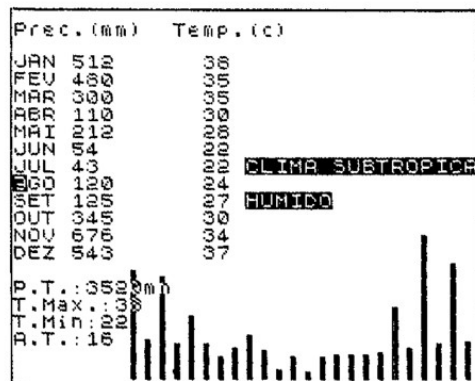


Fig. 44

Mecânica celeste

Com o programa que se apresenta a seguir (e que funciona no TS 1500, no ZX Spectrum e no TC 2068), podem resolver-se problemas de mecânica celeste.

A visualização animada de três corpos actuando gravitacionalmente entre si, no espaço, não é fácil de resolver com o recurso à álgebra, mas pode sê-lo com o computador.

O programa, tal como se encontra, destina-se a planetas. Todavia, bastarão pequenas alterações nas linhas 130, 150, 390, 400 e 410 para poder calcular — e mostrar de forma animada — por exemplo, a largada de um foguetão.

Para os entendidos, a base do programa é o teorema de Taylor:

$$x(t-dt) - x(t) - dt \cdot x'(t) - \frac{1}{2} (dt)^2 \cdot x''(t) - \frac{1}{6} (dt)^3 \cdot x'''(t) -$$

onde $x(t)$ é a velocidade no tempo t ; $x'(t)$ é a aceleração no tempo t ; $x''(t)$ é a mudança de aceleração média no tempo t , etc.

```

100 REM sistema solar
110 REM traca orbitas satelites
calculando as forcas gravitacio
nais muiuas
120 LET N=3
130 LET G=1
140 LET DT=0.20
150 DIM X(5): DIM Y(5)
160 DIM P(5): DIM Q(5)
170 DIM R(5): DIM S(5)
180 DIM M(5)
190 FOR I=1 TO N
200 READ M(I)
210 NEXT I
220 DATA 10000,100,100
230 FOR I=1 TO N
240 READ X(I),Y(I)
250 NEXT I
260 DATA 128,83
270 DATA 178,83
280 DATA 168,83
290 FOR I=1 TO N
300 READ UX,IY
310 LET P(I)=X(I)-UX*DT: LET Q(I)=Y(I)-IY*DT
320 NEXT I
330 DATA 0,-0.267
340 DATA 0,15.7
350 DATA 0,11
360 FOR I=1 TO N
370 LET R(I)=0: LET S(I)=0
380 FOR J=1 TO N
390 IF I=J THEN GO TO 520
400 LET DX=X(I)-X(J): LET DY=Y(I)-Y(J)
410 LET R=SQRT(DX*DX+DY*DY)
420 LET T=-G*M(I)*M(J)/R/R
430 LET R(I)=R(I)+T*DX/R: LET S(I)=S(I)+T*DY/R
440 NEXT J
450 NEXT I
460 FOR I=1 TO N
470 LET XF=2*X(I)-P(I)+DT*DT*R(I)/M(I)
480 LET YF=2*Y(I)-Q(I)+DT*DT*S(I)/M(I)
490 IF X(I)>255 THEN GO TO 530
500 IF X(I)<0 THEN GO TO 530
510 IF Y(I)>175 THEN GO TO 530
520 IF Y(I)<0 THEN GO TO 530
530 PLOT X(I),Y(I)
540 LET P(I)=X(I): LET Q(I)=Y(I)
550 LET X(I)=XF: LET Y(I)=YF
560 NEXT I
570 GO TO 420

```


Educação Musical

Se há coisa que o equipamento electrónico faz bem, desde que correctamente «afinado», é reproduzir sons. Assim, e atendendo até a uma certa falta de verbas para a aquisição de todos os instrumentos musicais que seriam necessários para o ensino desta matéria, o computador pode ser utilizado, com proveito, para apoio a algumas explicações no âmbito desta disciplina.

O programa seguinte, para o ZX Spectrum, é elementar e consiste, no essencial, em transformar o teclado do computador num teclado de piano. De qualquer modo, serve para iniciar na utilização do som:

```

10 REM musica
20 DIM x(50): DIM y(50)
30 LET k=0: LET l=1
40 BORDER 2: PAPER 4: INK 9: C
LS
50 PRINT AT 0,10: INVERSE 1: "T
OCANDO MUSICA"
60 PRINT "Escreva a sua musi
ca como uma linha de letras e
numeros."
70 PRINT "Indique as notas c
omo letras seguidas pelo seu
tempo em num."
80 PRINT "Dispoe de duas oit
avas. A mais baixa de A a G e a
mais alta de a a c."
90 INPUT n$
100 FOR a=1 TO LEN n$ STEP 2
110 IF CODE n$(a) < 97 THEN GO TO
280
120 IF n$(a) = "a" THEN LET k=-0.
S
130 IF n$(a) = "d" THEN LET k=0.5
140 IF n$(a) = "e" THEN LET k=1
150 IF n$(a) = "f" THEN LET k=1
160 IF n$(a) = "g" THEN LET k=1.5
170 LET y(l) = (CODE n$(a) - 67) + (2
*k)
180 LET l=l+1
190 LET k=0
200 NEXT a
210 LET l=1
220 FOR t=2 TO LEN n$ STEP 2
230 LET x(l) = VAL n$(t) / 2
240 LET l=l+1
250 NEXT t
260 GO TO 350
270 STOP
280 IF n$(a) = "A" THEN LET k=-0.
S
290 IF n$(a) = "B" THEN LET k=-0.
S
300 IF n$(a) = "F" THEN LET k=0.5
310 IF n$(a) = "G" THEN LET k=0.5

```

```

320 LET y(l) = (CODE n$(a) - 67 - k) *
2
330 LET k=0
340 LET l=l+1
350 GO TO 200
360 FOR z=1 TO LEN n$/2
370 BEEP x(z)/2, y(z)
380 NEXT z
400 INPUT "Quer tentar outra?";
i$
410 IF i$ = "s" OR i$ = "S" THEN RU
N

```

Este programa — como todos os que utilizem som — não pode ser utilizado pelo TS 1500. Este computador não tem circuito de som.

Mas é um programa que pode ser muito melhorado para utilização no TC 2068, uma vez que este possui quatro canais de som — característica que permite, portanto, a composição de acordes. Bastará, para tanto, utilizar o comando SOUND, com as respectivas variáveis indicadas no manual do TC 2068.

O programa seguinte, já um pouco mais avançado (mas igualmente simples de introduzir), serve para representar uma escala musical, podendo ser escritas notas musicais na mesma. Terminada a operação, pode imprimir-se na impressora.

No ZX Spectrum, bem como no TC 2068, pode introduzir-se uma rotina de som, por forma a que o computador produza o som correspondente à nota escrita, em órgão, piano ou outro instrumento.

```

10 REM escrever musica
30 LET x=151
50 FOR y=1 TO 5
50 PLOT 0,x: DRAW 255,0
70 LET x=x-8
80 NEXT y
90 FOR a=0 TO 7
100 READ t
110 POKE USR "l"+a,t
120 NEXT a
130 FOR b=0 TO 7
140 READ t
150 POKE USR "k"+b,t
160 NEXT b
165 REM Na linha 170 as letras
L e K são introduzidas em modo g
rafico
170 PRINT OVER 1; AT 3,3: "L"; AT
4,3: "K"
180 BEEP 0.25,0
200 DATA 0,4,7,5,4,4,4,4
220 DATA 0,60,124,124,124,56,0,
0
230 STOP

```

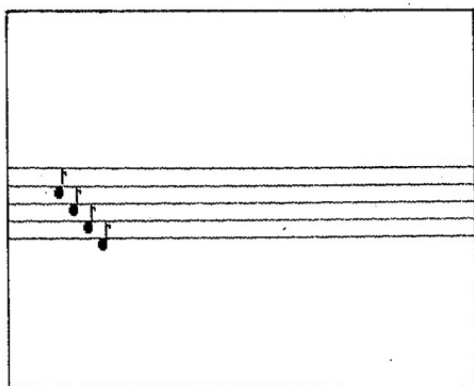



Fig. 45-A

Piano

Apesar de ser apenas um programa básico — mas assim mesmo um pouco longo —, não quis deixar de apresentar uma possibilidade de utilizar o teclado do Spectrum (ou TC 2068) como se de um piano se tratasse. Utiliza 5 oitavas, apesar de no écran surgir apenas uma oitava de cada vez (a mudança, para cima e para baixo, efectua-se premindo a tecla X).

É evidente que o programa é muito curto pelo que pode facilmente ser muito melhorado, por forma, por exemplo, a transformá-lo em outros instrumentos ou ainda a escrever as notas musicais à medida que as mesmas vão sendo tocadas (ver programa anterior).

Com o TC 2068 — e algumas modificações (que deverão ser executadas após consulta ao manual daquele computador) — é possível tirar um muito melhor rendimento do programa (não esquecer que o 2068 tem 4 canais de som, além de comandos para o «ataque», «decay», etc.).

```

5 LET C=0
10 REM Piano
20 INK 7: BORDER 0: PAPER 2: C
LS
30 PRINT AT 5,12: INVERSE 1: "P
IANO"
40 PRINT AT 8,0: "AS TECLAS BRA
NCAS SAO AS QUE VAO DE 'O' at
e 'P'."

```

```

50 PRINT AT 12,0: "PREMINDO 'Z'
OBTEN UMA HIGHER OITAVA, PRE
MINDO 'X' TERA UMA LOWER OITAVA"
60 PRINT AT 15,0: INVERSE 1: "P
RIMA A TECLA 'U' PARA VIBRATO
E 'M' PARA DESLIGAR"
70 PRINT AT 19,0: INVERSE 1: "P
RESSING 'C' PUTS MID C ON 'T' AG
AIN"
80 LET K=0: LET X=0.3
90 PAUSE 500
95 CLS
120 PRINT: INVERSE 1: AT 13,4: "
0": AT 13,6: "W": AT 13,8: "E": AT 13
10: "R": AT 13,12: "T": AT 13,14: "Y
": AT 13,16: "U": AT 13,18: "I": AT 1
3,20: "O": AT 13,22: "P"
122 PRINT AT 17,4: "F S L S D R
M F S L": AT 18,4: "A O A I O E I
A O A": AT 19,6: "L": AT 19,20: "L"
130 PRINT PAPER 0: AT 11,5: 2: AT
11,7: 3: AT 11,9: 4: AT 11,13: 6: AT 1
1,15: 7: AT 11,19: 9: AT 11,21: 0
132 FOR N=28 TO 200 STEP 16: PL
OT N,50: DRAW 0,25: NEXT N
134 FOR N=36 TO 180 STEP 16: PL
OT N,75: DRAW 0,27: NEXT N
135 PLOT 36,75: DRAW 48,0: PLOT
100,75: DRAW 32,0: PLOT 148,75:
DRAW 32,0
136 PLOT 28,75: DRAW 0,27: PLOT
92,75: DRAW 0,27: PLOT 140,75:
DRAW 0,27: PLOT 188,75: DRAW 0,2
7
137 PLOT 28,50: DRAW 160,0
138 PLOT 28,102: DRAW 160,0
139 FOR N=28 TO 200 STEP 16: PL
OT N,14: DRAW 0,24: NEXT N
140 IF INKEY$="Z" THEN LET K=12
150 IF INKEY$="X" THEN LET K=-1
2
160 IF INKEY$="C" THEN LET K=0
170 IF INKEY$="V" THEN LET X=0.
03
180 IF INKEY$="M" THEN LET X=0.
3
190 IF INKEY$="2" THEN POKE 228
85,41: BEEP X,-6+K: POKE 22885,7
200 IF INKEY$="3" THEN POKE 228
87,41: BEEP X,-4+K: POKE 22887,7
210 IF INKEY$="4" THEN POKE 228
89,41: BEEP X,-2+K: POKE 22889,7
220 IF INKEY$="6" THEN POKE 228
93,41: BEEP X,1+K: POKE 22893,7
230 IF INKEY$="7" THEN POKE 228
95,41: BEEP X,3+K: POKE 22895,7
240 IF INKEY$="9" THEN POKE 228
99,41: BEEP X,6+K: POKE 22899,7
250 IF INKEY$="0" THEN POKE 229
01,41: BEEP X,8+K: POKE 22901,7

```

```

260 IF INKEY$="q" THEN POKE 229
48,41: BEEP X,-7+k: POKE 22948,7
270 IF INKEY$="w" THEN POKE 229
50,41: BEEP X,-5+k: POKE 22950,7
280 IF INKEY$="e" THEN POKE 229
52,41: BEEP X,-3+k: POKE 22952,7
290 IF INKEY$="r" THEN POKE 229
54,41: BEEP X,-1+k: POKE 22954,7
300 IF INKEY$="t" THEN POKE 229
56,41: BEEP X,0+k: POKE 22956,7
310 IF INKEY$="y" THEN POKE 229
58,41: BEEP X,2+k: POKE 22958,7
320 IF INKEY$="u" THEN POKE 229
60,41: BEEP X,4+k: POKE 22960,7
330 IF INKEY$="i" THEN POKE 229
62,41: BEEP X,5+k: POKE 22962,7
340 IF INKEY$="o" THEN POKE 229
64,41: BEEP X,7+k: POKE 22964,7
350 IF INKEY$="p" THEN POKE 229
66,41: BEEP X,9+k: POKE 22966,7
355 IF INKEY$="k" THEN LET K=K+
0.1
355 IF INKEY$="l" THEN LET K=K+
0.1
360 GO TO 140
370 STOP

```

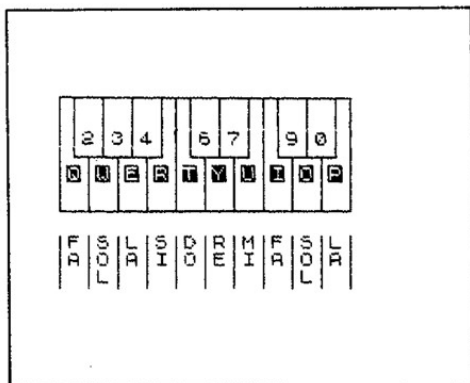


Fig. 45-B

Economia

O programa que se lista para Economia — ESTATÍSTICA —, servindo embora basicamente para esta disciplina (e não apenas a nível do Ensino Secundário), poderá ser aplicado a necessidades de qualquer outra, bem como a estudos sobre insucesso escolar, por exemplo, ou

outros. Por estas razões, trata-se de um programa mais longo, se bem que igualmente mais completo e elaborado.

Como se pode ver no menu principal, podem ser introduzidos dados; gerados dados aleatórios para estudos sobre o processo de funcionamento; efectuados cálculos estatísticos e histogramas; e ainda guardada ou carregada informação em suporte magnético.

NUMERO DE VALORES: 15			
DADOS FRACCION./INTEIROS f/i			
VALOR MINIMO	10		
VALOR MAXIMO	100		
valor do dado	1	=	1
valor do dado	2	=	8
valor do dado	3	=	40
valor do dado	4	=	72
valor do dado	5	=	26
valor do dado	6	=	18
valor do dado	7	=	19
valor do dado	8	=	13
valor do dado	9	=	64
valor do dado	10	=	7
valor do dado	11	=	61
valor do dado	12	=	86
valor do dado	13	=	55
valor do dado	14	=	51
valor do dado	15	=	70

Fig. 45-C

Pode introduzir-se (ou pedir-se) o número de valores que se quiser, fornecendo o cálculo respectivo o máximo, mínimo, média, variância e desvio standard.

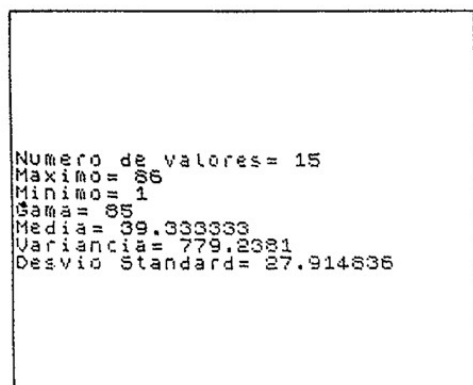


Fig. 45-D

Em qualquer momento se pode pedir listagem dos dados introduzidos, bem como alterar, apagar ou acrescentar qualquer deles:

```
N. de valores 10
Valor 1 = 12
Valor 2 = 23
Valor 3 = 45
Valor 4 = 123
Valor 5 = 77
Valor 6 = 5
Valor 7 = 67
Valor 8 = 208
Valor 9 = 208
Valor 10 = 1
Numero de valores = 10
Maximo = 123
Minimo = 1
Gama = 122
Media = 45.3
Variancia = 1928.6778
Desvio Standard = 43.916714
```

Fig. 45-E

Eis um exemplo do tipo de histogramas que o programa produz, a partir dos valores estatísticos calculados:

```
Valor maximo = 123
Valor minimo = 13
13
20.33
27.56
35
42.33
49.66
57
64.33
71.66
79
86.33
93.66
101
108.33
115.66
```

Fig. 46

A listagem do programa encontra-se no apêndice A.

Geometria

Pelas mesmas razões apontadas para a Educação Visual, também a Geometria (e outras áreas da Matemática) pode tirar todo o rendimento das capacidades gráficas dos computadores.

Para além do facto de programas de uma das disciplinas poderem servir a outra (ou algumas rotinas), apresentaremos aqui alguns programas curtos específicos desta matéria.

O primeiro destina-se a permitir o traçado de ângulos, chamando-se a atenção para a utilização das funções Seno, Coseno e ATN (arco tangente), entre outras matemáticas que o computador possui:

```
8800 REM angulos
8801 REM in-AX,AY
8802 REM out-THETA
8803 REM THETA e o angulo
8810 IF ABS AX>0.00001 THEN GO T
0 8860
8820 LET THETA=PI/2
8830 IF AY<0 THEN LET THETA=THET
A+PI
8840 IF ABS AY<0.00001 THEN LET
THETA=0
8850 RETURN
8860 LET THETA=ATN (AY/AX)
8870 IF AX<0 THEN LET THETA=THET
A+PI
8880 RETURN
```

Áreas e fórmulas

O programa que segue destina-se à apresentação das fórmulas para cálculo de áreas de figuras geométricas, apresentando, portanto, a fórmula e o cálculo de cada uma, quando a tal solicitado.

```
3 CLS
3 PRINT AT 1,7;"A AREA DE UM"
5 PRINT AT 3,5;"QUADRADO/RECT
ANGULO (1)"
10 PRINT AT 5,5;"TRIANGULO
(2)"
20 PRINT AT 7,5;"CIRCULO
(3)"
30 PRINT AT 9,5;"SEMI-CIRCULO
(4)"
40 PRINT AT 11,5;"ANEL
(5)"
50 PRINT AT 13,5;"ESCOLHA 1 a
5"
60 LET g$=INKEY$
70 IF g$="" OR g$<"1" OR g$>"5
" THEN GO TO 60
80 GO SUB VAL g$*1000
90 GO TO 2
1000 CLS
1010 PRINT "INDIQUE COMPRIMENTO:
"
1020 INPUT L
1030 PRINT L
1040 PRINT "INDIQUE LARGURA: ";
```

```

1050 INPUT b
1050 PRINT b
1051 PRINT AT 4,5;"ESTA CORRECTO
? (s/n)"
1052 IF INKEY$="n" THEN GO TO 10
00
1053 IF INKEY$<>"s" THEN GO TO 1
052
1055 CLS
1057 PRINT "FORMULA PARA A AREA
DE UM", "QUADRADO/RECTANGULO E:",
" (COMPRIMENTO*LARGURA)"
1058 PRINT
1070 LET a=l*b
1080 PRINT "QUADRADO/RECTANGULO:"
", "COMPRIMENTO ";l;TAB 0;"LARGUR
A ";b;TAB 0;"AREA ";a
1085 PRINT AT 15,5;"PRIMA QUALQU
ER TECLA"
1090 PAUSE 4e4
1100 RETURN
2000 CLS
2010 PRINT "INDIQUE ALTURA: ";
2020 INPUT h
2030 PRINT h
2040 PRINT "INDIQUE BASE: ";
2050 INPUT b
2060 PRINT b
2061 PRINT AT 4,5;"ESTA CORRECTO
? (s/n)"
2062 IF INKEY$="n" THEN GO TO 20
00
2063 IF INKEY$<>"s" THEN GO TO 2
062
2065 CLS
2067 PRINT "FORMULA PARA A AREA
DE UM", "TRIANGULO E (BASE*ALTURA
)/2"
2068 PRINT
2070 LET a=b*h/2
2080 PRINT "TRIANGULO:", "ALTURA
";h;TAB 0;"BASE ";b;TAB 0;"AREA
";a
2085 PRINT AT 15,5;"PRIMA QUALQU
ER TECLA"
2090 PAUSE 4e4
2100 RETURN
3000 CLS
3010 PRINT "INDIQUE O RAIO: ";
3020 INPUT r
3030 PRINT r
3040 PRINT AT 4,5;"ESTA CORRECTO
? (s/n)"
3050 IF INKEY$="n" THEN GO TO 30
00
3050 IF INKEY$<>"s" THEN GO TO 3
050
3070 CLS
3080 PRINT "FORMULA PARA A AREA
DE UM", "CIRCULO E:", "(PI*(RAIO*

```

```

RAIO))"
3090 PRINT
3091 LET a=PI*(r*r)
3092 PRINT "CIRCULO:", "RAIO ";r
;TAB 0;"PI ";PI;TAB 0;"AREA ";a
3093 PRINT AT 15,5;"PRIMA QUALQU
ER TECLA"
3094 PAUSE 4e4
3100 RETURN
4000 CLS
4010 PRINT "INDIQUE RAIO: ";
4020 INPUT r
4030 PRINT r
4040 PRINT AT 4,5;"ESTA CORRECTO
? (s/n)"
4050 IF INKEY$="n" THEN GO TO 40
00
4050 IF INKEY$<>"s" THEN GO TO 4
050
4070 CLS
4080 PRINT "FORMULA PARA A AREA
DE UM", "SEMI-CIRCULO E:", "(PI*(
RAIO*RAIO)/2)"
4090 PRINT
4091 LET a=PI*(r*r)/2
4092 PRINT "SEMI-CIRCULO:", "RAI
O ";r;TAB 0;"PI ";PI;TAB 0;"AREA
";a
4093 PRINT AT 15,5;"PRIMA QUALQU
ER TECLA"
4094 PAUSE 4e4
4100 RETURN
5000 CLS
5010 PRINT "INDIQUE RAIO EXTERIO
R: ";
5020 INPUT o
5030 PRINT o
5040 PRINT "INDIQUE RAIO INTERIO
R: ";
5050 INPUT i
5060 PRINT i
5062 IF o<=i THEN PRINT "O ANEL
NAO EXISTE": PAUSE 80: GO TO 2
5070 PRINT AT 5,4;"ESTA CORRECTO
? (s/n)"
5072 IF INKEY$="n" THEN GO TO 50
00
5073 IF INKEY$<>"s" THEN GO TO 5
072
5075 CLS
5077 PRINT "FORMULA DA AREA DE U
M ANEL: (PI*(RAIO EXT.*RAIO
EXT.))- (PI*(RAIO INT.*RAIO
INT.))"
5080 LET b=o*o
5081 LET c=i*i
5082 LET a=PI*(b-c)
5090 PRINT "ANEL:", "RAIO EXTE
RIOR ";o;TAB 0;"RAIO INTERIOR ";
i;TAB 0;"AREA=";a

```

```

5095 PRINT AT 15,5;"PRIMA QUALQU
ER TECLA"
5097 PAUSE 4e4
5100 RETURN

```

```

FORMULA DA AREA DE UM ANEL:
(PI*(RAIO EXT.*RAIO EXT.))-
(PI*(RAIO INT.*RAIO INT.))

```

```

ANEL
RAIO EXTERIOR 50
RAIO INTERIOR 30
AREA=5026.5462

```

PRIMA QUALQUER TECLA

Fig. 47

Matemática (e outras)

O programa que escolhemos para a matemática (e só escolhemos um porque, ao fim e ao cabo, o computador funciona, em qualquer programa, com princípios matemáticos) destina-se a representar funções lineares em gráficos tridimensionais.

A função matemática que determina o gráfico encontra-se na linha 1010, podendo ser alterada para qualquer valor que se pretenda.

Quando se faz RUN, o computador pergunta qual a resolução. Trata-se de um factor que determina o espaço entre os pontos do desenho; quanto menor for o número, melhor será a resolução (mas também mais tempo levará a execução do gráfico).

De qualquer modo, damos a seguir exemplos de várias funções com diferentes resoluções:

Considere-se a função
 $t = -\text{EXP } t/80$

representada com uma resolução de 5 espaços:

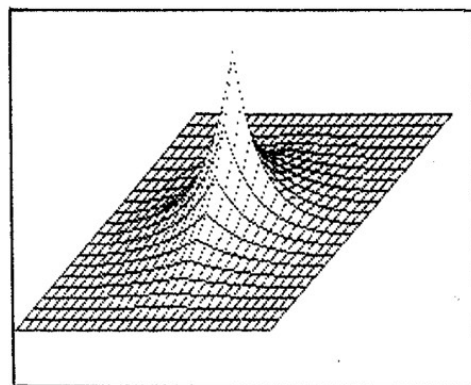


Fig. 48

Compare-se agora a função

$$t = \text{COS}(T*4)/6$$

representada com uma resolução de 5 espaços

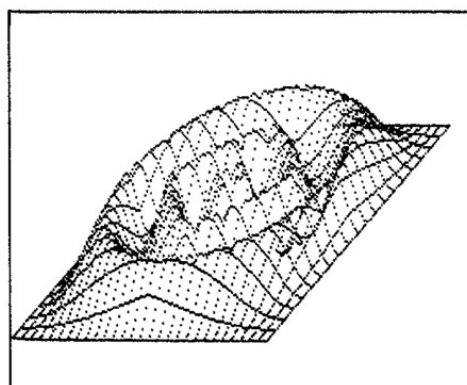


Fig. 49

e a mesma função representada com uma resolução de 2 espaços

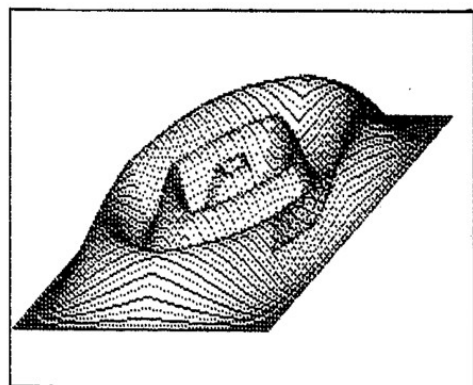


Fig. 50

Atentemos agora no caso da função

$$t = (\sin(t * 3) + \cos t) / 4$$

Com uma resolução de 10 espaços, a figura quase perde a sua característica tri-dimensional.

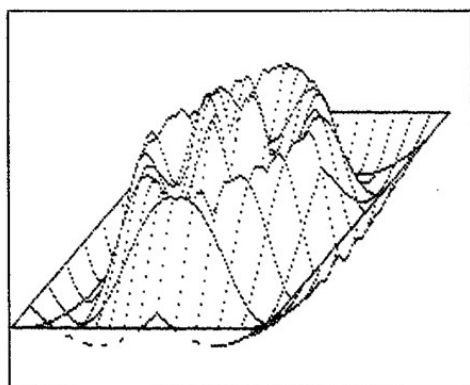


Fig. 51

Mas com uma resolução de 2 espaços, quase se torna um desenho em relevo compacto

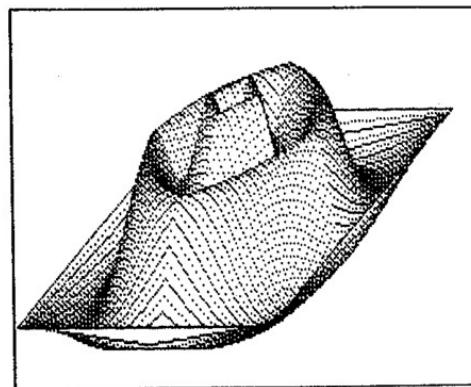


Fig. 52

Eis o programa — simples, como se pode ver — que permite a realização dos gráficos apresentados:

```

1 BORDER 0: PAPER 0: CLS
2 INPUT "resolucao": s
3 DIM p(250,2)
4 FOR f=1 TO 250: LET p(f,2)=
255: IF f>140 THEN LET p(f,2)=f
12 NEXT f: BEEP .5,30
15 FOR f=-50 TO 50 STEP s
20 LET a=f
25 LET b=50-ABS f
30 FOR g=-70 TO 70
35 LET c=70-ABS g
40 GO SUB 1000
50 NEXT g
55 IF f=50 THEN STOP
60 FOR a=f+1 TO f+s-1
65 LET b=50-ABS a
70 FOR g=-70 TO 70 STEP s
75 LET c=70-ABS g
80 GO SUB 1000
90 NEXT g
100 NEXT a
110 NEXT f
1000 LET t=b*c/800
1005 LET r=a+g+121
1010 LET t=cos(t*4)/5
1020 LET t=INT(80+a-t*80)
1022 IF f=-50 THEN LET p(r,2)=t
1023 IF t<=p(r,1) THEN GO TO 105
0

```

```

1025 INK 5
1026 LET P(r,1)=t
1027 IF t<0 THEN LET t=0
1028 IF t>175 THEN LET t=175
1035 PLOT r,t
1040 RETURN
1050 IF t>=P(r,2) THEN RETURN
1060 LET P(r,2)=t
1070 INK 5
1080 GO TO 1027

```

Trabalhos Oficiais (Educação Visual)

A disciplina de Trabalhos Oficiais tem, habitualmente e por razões que não cabe analisar nesta obra, o estatuto de menos importante, sendo relegada, em vários aspectos, para um plano secundário. Não só por isso, entendi dever incluí-la no presente livro, tanto mais que o computador pode revelar-se, naturalmente, um precioso auxiliar de ensino nas matérias ministradas ao abrigo dos programas de T.O.

Existem programas diversos que permitem desenho de peças (os quais se adaptam ao ensino técnico), com aplicação na electricidade, electrónica e outras áreas, programas esses cuja indicação será dada no capítulo dedicado aos comerciais.

Na área das tapeçarias (e outras), contudo, é possível, com rotinas muito simples e fáceis de introduzir no computador, produzir padrões que poderão servir quer para estudos quer para posterior execução, já que podem ser impressos em papel e guardados em periféricos (gravador, microdrives ou diskettes).

A título de exemplo, apresentamos um programa curto, com o qual é possível, mediante a alteração de algumas linhas apenas, produzir um número infinito de padrões, simétricos ou não. O programa tem uma pausa na linha 750, o que permite interrompê-lo num dado momento do desenvolvimento do seu padrão, imprimi-lo (ou estudá-lo) e prosseguir com a instrução CONTINUE.

A seguir se apresenta uma sequência (não completa) de padrões produzidos a partir do programa tal como se encontra «listado».

```

15 BORDER 2: PAPER 2: INK 7: C
LS
20 DRAW 255,0: DRAW 0,175: DRA
W -255,0: DRAW 0,-175
25 PRINT AT 9,7: "PROGRAMA PADR
DES"
35 PLOT 0,23: DRAW 255,0
40 PRINT AT 20,1: " ";

```

```

42 RESTORE
45 FOR X=1 TO 23
50 READ Logo
55 PRINT CHR$(Logo); BEEP 0.05
,Logo/2
60 NEXT X
65 DATA 32,32,32,32,127,32,32,
76,85,73,83,32,32,100,101,32,32,
67,65,77,80,79,83
70 PRINT AT 12,9;"
75 PRINT #0;AT 1,4;"Press. tec
la p/continuar"
80 PAUSE 0
90 BORDER 1: PAPER 1: INK 6: C
LS
100 FOR x=-127 TO 128 STEP 2
110 PLOT OVER 1;127,0
120 DRAW OVER 1;x,175
130 NEXT x
140 GO SUB 750
200 FOR x=-127 TO 128 STEP 2
210 PLOT OVER 1;127,175
220 DRAW OVER 1;-x,-175
230 NEXT x
240 GO SUB 750
300 FOR y=88 TO -87 STEP -2
310 PLOT OVER 1;0,87
320 DRAW OVER 1;255,y
330 NEXT y
340 GO SUB 750
400 FOR y=88 TO -87 STEP -2
410 PLOT OVER 1;255,87
420 DRAW OVER 1;-255,y
430 NEXT y
440 GO SUB 750
500 FOR y=175 TO 0 STEP -2
510 PLOT OVER 1;0,y
520 DRAW OVER 1;255,0
530 NEXT y
540 GO SUB 750
600 FOR x=0 TO 255 STEP 1
610 PLOT OVER 1;x,0
620 DRAW OVER 1;0,175
630 NEXT x
640 GO SUB 750
700 GO TO 100
750 FOR z=0 TO 300
760 NEXT z
770 RETURN

```

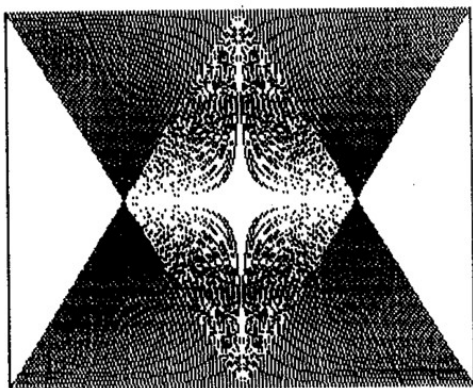


Fig. 53

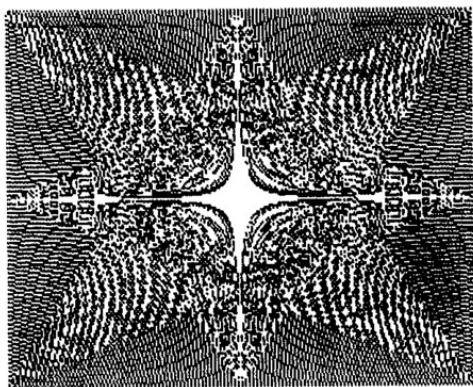


Fig. 54

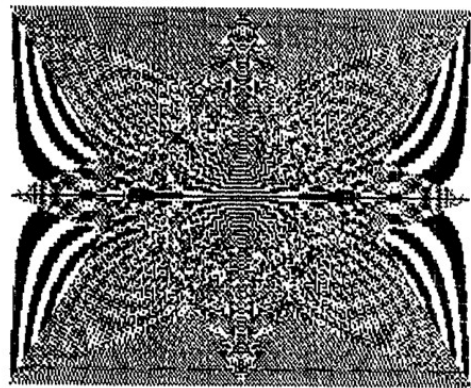


Fig. 55

O programa que se segue destina-se à produção dos conhecidos padrões Moiré:

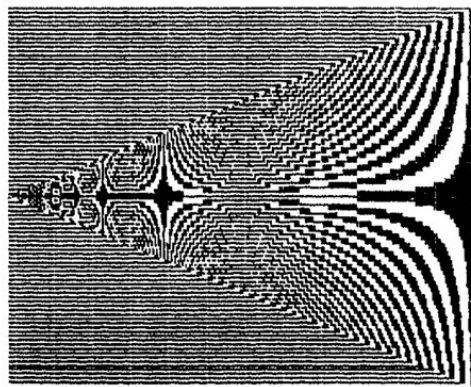


Fig. 56


```

LS 1 PAPER 0: INK 7: BORDER 0: C
5 REM padroes Moire
5 REM by
7 REM *****
10 FOR x=0 TO 255
20 PLOT x,0
30 DRAW OVER 1;255-x*2,175
40 NEXT x
50 FOR y=0 TO 175
60 PLOT 0,y
70 DRAW OVER 1;255,175-y*2
80 NEXT y

```

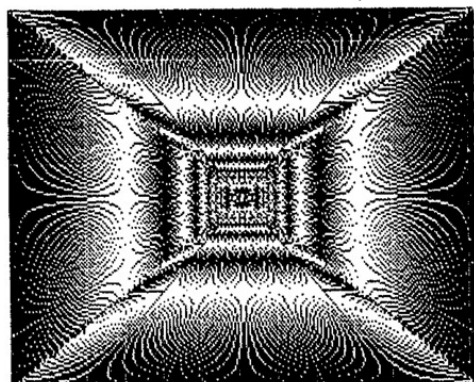


Fig. 57

Avaliação

Um dos problemas (sobretudo de tempo, mas não só) que mais aflige os professores, é o da avaliação. É claro que o computador não pode —a não ser em testes do tipo americano —«ver» os pontos, em lugar do professor. Isto implicaria tais condicionantes que, na prática, tornariam impossível essa tarefa.

Todavia, a informática pode constituir poderoso auxiliar na memorização de resultados; cálculo de médias; cálculo de totais; e até, caso lhe sejam fornecidos factores constantes, resultados finais. Tudo isto, como é óbvio, soluciona alguns importantes problemas de «enganos» e tempo.

O programa que se apresenta constitui sobretudo uma base para desenvolvimento. De qualquer modo, e assim mesmo como se encontra, permite a indicação do número de alunos da turma, a disciplina, a data, o número de questões e, a partir da introdução dos dados, não só os memoriza como os imprime numa tabela, calculando simultaneamente

os resultados. Estes poderão ser impressos em papel e guardados em caderneta.

Com pequenos acrescentos, poderá ainda fazer um cálculo médio dos resultados por questão, bem como a memorização de cada resultado. Caso se pretenda ligá-lo ao do ficheiro de alunos, será possível guardar todos os resultados no referido ficheiro.

```

2000 REM tabela de correccao
2050 LET GT=0
3000 LET a=0
3500 LET b=0
4000 LET T=0
4500 LET P=0
5000 DIM A(30,15)
1000 INPUT "Qual o teste?";A$
1050 INPUT "Qual o ano?";B$
1100 INPUT "Qual a turma?";C$
1150 INPUT "Qual a data?";D$
1200 INPUT "Qual a data?";E$
1250 INPUT "Qual a data?";F$
1300 INPUT "Numero de alunos?";M
1350 INPUT "Numero de perguntas?";N
1500 GO SUB 6000
2000 FOR Q=1 TO M
2100 INPUT "Aluno numero";a
2200 IF a>M THEN GO TO 2100
2300 PRINT OVER 1;AT a+3,0;a
2400 FOR m=1 TO N
2500 INPUT "Questao numero ";b
2600 IF b>N THEN GO TO 2500
2700 INPUT "Cotacao atribuida";C
2800 LET A(a,b)=C
2900 PRINT OVER 1;AT a+3,b*2;C
3000 LET T=T+C
3100 LET GT=GT+C
3200 NEXT m
3300 PRINT OVER 1;AT a+3,30;T
3400 LET T=0
3500 NEXT Q
5000 REM ***desenho do quadro***
5020 PRINT " * TESTE DE ";A$;"
*
6030 PLOT 0,175: DRAW 255,0
6050 PRINT B$;"ANO";" ";D$;"TURMA";C$;" ";E$;" ";F$
6100 PLOT 0,159: DRAW 255,0
6150 PLOT 0,175: DRAW 0,-175
6200 PLOT 255,175: DRAW 0,-175
6250 PLOT 0,152: DRAW 255,0
6300 PRINT OVER 1;AT 2,0;" *
TABELA DE CORRECCAO *
6350 PRINT OVER 1;AT 3,0;"a"
6400 FOR b=1 TO N
6450 PRINT AT 3,b*2;b
6500 NEXT b
6550 PRINT OVER 1;AT 3,30;"T."
6600 PLOT 0,144: DRAW 255,0

```

```

6700 FOR q=15 TO b*17 STEP 16
6800 PLOT q,152: DRAW 0,-152
6900 PLOT 240,152: DRAW 0,-152
7000 NEXT q
7100 RETURN

```

* TESTE DE Inglês *									
10. ANO		TURMA H						10.9.85	
* TABELA DE CORRECCAO *									
Aluno	1	2	3	4	5	6	7	8	T
1	5	0	11	7	10	9	3	4	49
2	10	9	13	8	8	9	5	4	67
3	10	4	7	10	10	7	5	5	64
4	0	0	9	7	10	9	5	5	55
5	0	0	10	9	7	7	5	5	53
6	0	0	5	5	9	10	4	5	48
7	0	0	5	5	9	10	4	5	48
8	0	0	5	5	9	10	4	5	48
9	0	0	5	5	9	10	4	5	48
10	0	0	5	5	9	10	4	5	48

Fig. 58

Seguidamente apresenta-se a rotina possível para o cálculo das médias, a qual pode ser aplicada, quer às notas de um aluno, quer às várias classificações de uma pergunta de um teste.

Nota 1	-	10
Nota 2	-	11
Nota 3	-	8
Nota 4	-	13
Nota 5	-	10
Nota 6	-	11
Nota 7	-	10.5
Nota 8	-	10.5
A media e	-	11

Fig. 58-A

```

10 REM © Luis de Campos 1985
20 REM Resultados medios
30 LET s=0
100 INPUT "Quantas notas? ";a
102 DIM m(a)
105 FOR i=1 TO a
110 PRINT AT 20,0;"Nota ";i
120 INPUT m(i)
130 PRINT AT i,0;"Nota ";i;" - ";m(i)
140 LET s=s+m(i)
150 NEXT i
160 LET t=s/a
170 PRINT "A media e ";t

```

3.2 — Algumas rotinas e programas úteis

No capítulo sobre a programação, falei da vantagem em fazê-la por blocos, uma vez que os mesmos poderiam posteriormente ser aproveitados para outros programas.

É o momento de serem apresentadas aqui pequenas rotinas ou blocos, cuja utilização é muito frequente para as mais variadas necessidades programáticas.

Renumeração das linhas do programa

Apesar do muito cuidado que possa haver ao projectar o programa, por vezes chega-se a um ponto em que é necessário espaço entre linhas, espaço esse já não existente. É nessa altura que uma rotina deste tipo se mostra particularmente útil.

Trata-se de uma rotina muito simples e que, por isso, não faz a renumeração das instruções GOSUB e GOTO, as quais terão de ser posteriormente corrigidas.

```

9010 PRINT "Numero de linhas a r.
enumerar?"; INPUT n: PRINT n
9020 PRINT "Numero da primeira l
inha?"; INPUT s: PRINT s
9030 PRINT "Espaco entre linhas?";
INPUT i: PRINT i: IF n=0 OR
s=0 OR i=0 THEN PRINT "Impossible
!!!": GO TO 9010
9040 LET b=PEEK 23635+256*PEEK 2
3635
9050 LET d=s: GO SUB 9900: FOR j
=1 TO n-1: LET b=b+4+PEEK (b+2)+
256*PEEK (b+3): LET d=d+i: GO SU
B 9900: NEXT j
9060 PRINT "Terminado. Liste o p
rograma."
9099 STOP
9900 LET dh=INT (d/256): LET dl=
d-256*dh: POKE b,dh: POKE b+1,dl
: RETURN

```

Como alternativa mais rápida (já que usa a função FN), apresenta-se a seguinte, que tem a vantagem de imprimir uma lista com a numeração antiga e a nova, facilitando assim a correção dos GOSUB e GOTO.

```

9700 LET P=FN P()
9710 INPUT "Número inicial ";SO
9720 INPUT "Espaço ";IO
9730 PRINT "ANT";TAB 10;"NOVO"
9740 LET L=FN L(P)
9750 IF L>9000 THEN STOP
9760 PRINT L;TAB 10;SO
9770 POKE P,INT (SO/256)
9780 POKE P+1,SO-INT (SO/256)*256
9790 LET SO=SO+IO
9800 LET P=P+4+PEEK (P+2)+256*PE
EK (P+3)
9810 GO TO 9740

```

Leitura de cabeças

Por vezes necessitamos saber o nome de um programa que se encontra gravado em cassette, bem como o seu tipo e ainda outras características. A rotina que se segue destina-se a esse fim.

```

10 DATA 17,17,0,175,55,221,33,
15,91,205,85,5,48,242,201
20 FOR a=23296 TO 23310
30 READ d
40 POKE a,d
50 NEXT a
60 LET a=USR 23296
70 PRINT "TIPO=";PEEK 23311;
80 PRINT "NOME=";
90 FOR i=1 TO 10
100 PRINT CHR$(PEEK (23311+i))
;
110 NEXT i
120 PRINT
130 GO TO 60

```

A rotina seguinte — que até pode ser acrescentada à anterior — serve para indicar, acerca de um programa gravado em cassette, a quantidade de memória dispendida pelo programa e suas variáveis, bem como — o que é mais importante — a memória ainda livre na RAM do computador.

```

9000 DEF FN P(N)=PEEK (N)+256*PE
EK (N+1)
9010 PRINT "Programa: ";FN P(236
27)-FN P(23635)
9020 PRINT "Variáveis: ";FN P(23
641)-FN P(23627)
9030 PRINT "Mem. livre: ";FN P(23
613)-FN P(23653)
9040 RETURN

```

Sobreimpressão

Um dos problemas que vulgarmente são postos à utilização do computador para processamento de texto — e ensino de línguas — é a não existência, no alfabeto informático normalizado, de cedilhas, acentos e outros sinais gráficos.

O BASIC do ZX Spectrum (e do TC 2068) possui o comando OVER, que permite sobreimprimir um carácter sobre outro previamente impresso. Só que... por razões um tanto complexas (e que não vem ao caso explicar aqui), o resultado nem sempre é o esperado; quero dizer que quando essa sobreimpressão se efectua com cruzamento de traços, por exemplo, o resultado é um novo carácter esquisito.

Para evitar tais problemas e procurar solucionar a sobreimpressão «real», apresentam-se duas rotinas.

A primeira utiliza o comando OVER, e destina-se fundamentalmente a sublinhados:

```

10 CLS
20 PRINT AT 5,5;"Levee"
30 PRINT AT 5,8;OVER 1;"_"
40 PRINT AT 7,15;"_"
50 PRINT AT 7,15;OVER 1;"-"
60 PRINT AT 10,11;"Sublinhado"
70 PRINT AT 10,11;OVER 1;"_"
;
80 PRINT AT 12,15;"■ ■"
90 PRINT AT 12,15;OVER 1;"_"

```

A que se apresenta a seguir, mais complexa, necessariamente, permite fazer sobreimpressão efectiva. (Não esqueça que deverá definir caracteres para os acentos, podendo recorrer por exemplo às posições livres das teclas dos números — quando em modo CAPS SHIFT.)

Para definição de caracteres há uma série de programas, quer em livros, quer em revistas. Apresentamos um neste capítulo.

```

10 PRINT AT 10,10;"X"
20 LET X=95
30 LET Y=99
40 LET C$="2"
50 GO SUB 5000
60 STOP
5000 LET a=256+PEEK 23606+256*PE
EK 23607
5010 LET a=a+8*(CODE C$-32)
5020 FOR i=0 TO 7
5030 LET d=PEEK (a+i)
5040 FOR j=0 TO 7
5050 LET b=d-INT (d/2)*2
5060 LET d=INT (d/2)
5070 IF b=1 THEN PLOT X,Y
5080 LET X=X-1
5090 NEXT j
5100 LET Y=Y-1
5110 LET X=X+8
5120 NEXT i
5130 RETURN

```

Diagonais

A rotina seguinte permite dispor títulos ou texto em diagonal, no écran:

```

10 LET a$="Uma mensagem"
20 LET X=10
30 LET Y=170
40 FOR k=1 TO LEN (a$)
50 LET c$=a$(k)
60 GO SUB 5000
70 LET X=X+4
80 NEXT k
90 STOP
5000 LET a=255+PEEK 23506+255*PE
5010 LET a=a+8*(CODE c$-32)
5020 FOR i=0 TO 7
5030 LET d=PEEK (a+i)
5040 FOR j=0 TO 7
5050 LET b=d-INT (d/2)*2
5060 LET d=INT (d/2)
5070 IF b=1 THEN PLOT X,Y
5080 LET X=X-1
5090 NEXT j
5100 LET Y=Y-1
5110 LET X=X+8
5120 NEXT i
5130 RETURN

```

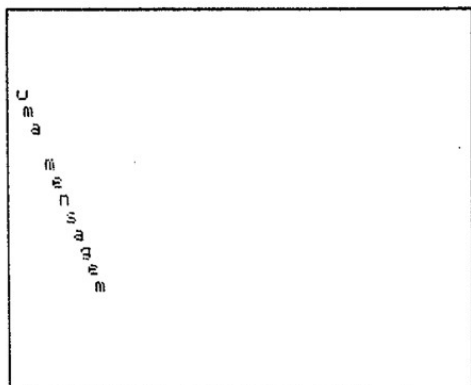


Fig. 58-B

Tamanhos

Por vezes há necessidade de utilizar letras e números de diversos tamanhos. Para tanto se apresenta uma rotina muito simples que poderá ser facilmente modificada por forma a produzir letras ou outros caracteres de um só tamanho:

```

10 LET a$="ABCDE"
20 LET X=30
30 LET Y=170
40 FOR k=1 TO LEN (a$)
50 LET c$=a$(k)
60 GO SUB 5000
65 LET X=INT (X+5X/2)
66 LET SX=SX-1
67 LET SY=SY-1
70 LET X=X+4
80 NEXT k
90 STOP
5000 LET a=255+PEEK 23506+255*PE
5010 LET a=a+8*(CODE c$-32)
5020 FOR i=0 TO 7
5030 LET d=PEEK (a+i)
5040 FOR j=0 TO 7
5050 LET b=d-INT (d/2)*2
5060 LET d=INT (d/2)
5070 IF b=1 THEN GO SUB 5000
5080 LET X=X-SX
5090 NEXT j
5100 LET Y=Y-SY
5110 LET X=X+5X+8
5120 NEXT i
5130 RETURN
6000 FOR m=1 TO SY
6010 FOR n=1 TO SX
6020 PLOT X+n,Y+m
6030 NEXT n
6040 NEXT m
6050 RETURN

```

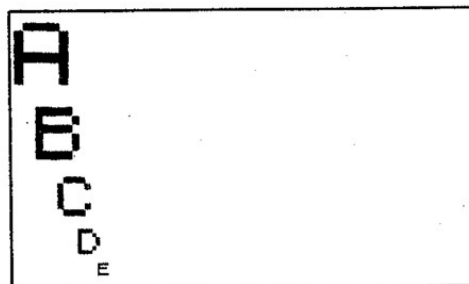


Fig. 58-C

Teste à RAM

Esta rotina destina-se a testar a qualidade (deficiências possíveis) das placas de circuitos integrados que constituem a memória de acesso alea-

tório (RAM) do computador. É uma rotina demorada, mas muito completa, já que testa *todos* os endereços da referida memória.

De qualquer modo, em princípio, só será utilizada quando se adquirir um novo computador. Após o aparecimento de diversas cores, no écran, aparecerá o número 225 a encher o dito écran, durante aproximadamente 15 minutos. Se, em qualquer momento, surgir um número diferente... então é porque um dos «chips» da RAM está avariado.

```
5 CLEAR 24199
10 FOR x=0 TO 7
20 PAPER x
30 CLS
40 PAUSE 20
50 NEXT x
60 FOR y=52269 TO 65530
70 POKE y,255
75 IF PEEK y<>255 THEN BEEP 5,4
80 PRINT PEEK y;" ";
85 POKE 23692,-1
90 NEXT y
```

Bases de dados

As bases de dados são, sem exagero, a razão de existir de todo o trabalho dos computadores. É nelas que se guarda, de forma sistematizada e acessível, toda a informação (nos modos mais diversificados), sendo ela igualmente quem fornece essa mesma informação, quer de forma simples, quer combinada com outras bases.

Assim, e apesar de existirem obras inteiramente dedicadas ao assunto — as quais deverão, de qualquer modo, ser consultadas a partir de certa evolução (e verificadas certas necessidades específicas) do programador — apesar de existirem tais obras, dizia, aqui fica um programa que permitirá iniciar o trabalho com tais bases. Atendendo às dificuldades sentidas, vulgarmente, para programar bases de dados destinadas a serem utilizadas com periféricos mais complexos — mas mais potentes — que as cassettes, bem como à simplicidade que implica programar uma base de dados para este tipo de periférico, optei por fornecer a listagem de um programa destinado a *microdrive*, o qual pode, também, ser muito facilmente aplicado a *diskette*, com ligeiras alterações.

Tal com se encontra, permite tipificar ficheiros com os seus diversos campos; alterá-los; apagá-los; gravar e carregar a informação. No programa se encontram as instruções necessárias à operação, bem como as explicações sobre os blocos.

```
10 REM © Luis Campos 1985
15 DIM h$(4,8): DIM e$(14,32)
20 LET n$="Introduza a wafer d
e dados"+CHR$(13)+"no Microdrive..."
30 LET n$="Veja a ultima linha
para"+CHR$(13)+"instruções."
40 LET j=1: LET y$="Responda (
s/n).": LET z$="Carregue numa te
cla..."
```

```
45 LET in=4: LET out=5
50 FOR n=1 TO 4: READ h$(n): N
EXT n
60 REM linhas 60 a 80 reservad
as a instruções
70 REM para printer tais como
Format "t";600 e Open #3;"t" con
forme
80 REM as necessarias para o s
eu printer.
90 DATA "Prim.", "Seg.", "Terc.",
"Quarto"
100 CLS: LET t$="BASE DADOS":
GO SUB 1000
110 PAUSE 50: PRINT "Precisa d
e instruções?";y$
120 GO SUB 1050: IF k$="s" OR k
$="S" THEN GO SUB 1100
130 CLS: LET t$="MENU": GO SUB
1000
140 PRINT "1.Inic.novo tipo de
ficheiro." "2.Inic.introdução e
m ficheiro." "3.Apagar, alterar o
u acrescentar items." "4.Lista
gem total de ficheiro." "5.Selec
ção de um item." "6.Terminar ou
guardar programa."
150 PRINT "Escolha o numero":
GO SUB 1050: LET k=VAL k$
155 IF k<1 OR k>6 THEN PRINT "A
penas entre 1 e 6...": GO TO 150
160 GO SUB 1800+200*k
170 CLS: PRINT "Quer voltar ao
Menu?"
180 PRINT: GO SUB 1050: IF k$="
s" OR k$="S" THEN GO TO 140
200 GO TO 9999: REM end
1000 PRINT TAB 15-LEN t$/2;t$: R
ETURN
1050 LET k$=INKEY$: IF k$="" OR
k$=CHR$(13) THEN GO TO 1050
1060 RETURN
1100 CLS: LET y$="INSTRUCCOES":
GO SUB 1000: PRINT
1105 PRINT "Este programa permit
e-lhe criar um certo numero de i
tems. Uma vez criadas, jun
tamente com um nome de ficheiro,
passa a dispor do esqueleto
de um pro- grama de Base de Dad
os."
1107 PRINT "Se, apos isso, utili
zar a opcao Fim e Gravar, do men
u, este con-junto de items apare
cera todas as vezes que utilize
o programa."
1110 PRINT "Uma vez gravado o se
u programa, pode carrega-lo e in
troduzir dados, ate encher a
wafer do Microdrive." "Uma ve
```

```

z gravados os dados, pode utiliz
a-los, adicionar, apagar, altera
r ou listar o ficheiro."
1115
1120 PRINT "z$": GO SUB 1050
1195 RETURN
1500 INPUT q$: IF LEN q$>32 THEN
PRINT "Demasiado longo...": GO
TO 1500
1505 RETURN
2000 CLS: LET t$="NOVO FICHEIRO
-ESPECIFICACOES": GO SUB 1000: P
AUSE 100
2005 PRINT "Selecione os seus
titulos""premiendo ENTER apos c
ada um.""So pode usar quatro ti
tulos!"
2010 FOR n=1 TO 4: PRINT h$(n):"
e:": GO SUB 1500: PRINT q$
2015 LET e$(n)=q$: NEXT n
2020 PRINT "Fim da especificacao
dos titulos""Agora o nome do f
icheiro""(maximo 10 caracteres)"
2025 INPUT "Nome do ficheiro: ";
f$
2030 IF LEN f$>10 THEN PRINT "De
masiado longo...""EU disse: max
imo 10 caracteres!": GO TO 2025
2040 RETURN
2200 CLS: LET t$="INTRODUCAO DE
ITEMS": GO SUB 1000: PAUSE 50:
GO SUB 5000: LET str=in
2205 PRINT "Items podem ser intr
oduzidos ate""introduzir ->X<-""
2210 PRINT "Lancamento n.": j$
2215 PRINT e$(1): INPUT q$: IF q
$="X" THEN GO TO 2240
2220 PRINT e$(2): INPUT r$
2225 PRINT e$(3): INPUT s$
2230 PRINT e$(4): INPUT u$
2235 PRINT #str;q$: PRINT #str;r
$;s$;u$
2235 LET j=j+1: GO TO 2210
2240 LET j=j-1: PRINT "Fim de in
tr.": PRINT #str;CHR$ 0: PAUSE 5
0: CLOSE #str
2245 RETURN
2400 PRINT "Opcoes:""1.Aumenta
r o ficheiro""2.Alterar um item
""3.Apagar um item""4.Voltar a
o Menu"
2405 PRINT "Escolha um numero":
GO SUB 1050: LET k=JAL k$: IF k
<1 OR k>4 THEN PRINT "So de 1 a
4...": GO TO 2405
2410 GO SUB 4000+100*k
2415 CLOSE #in: CLOSE #out
2420 RETURN
2500 CLS: LET t$="LISTAGEM DO F
ICHEIRO": GO SUB 1000: PAUSE 50

```

```

2605 PRINT "Quer imprimir no ec
ran""ou no printer"
2610 PRINT "P-printer; S-Ecran"
2615 GO SUB 1050: IF k$<>"p" AND
k$<>"s" THEN PRINT "Deve escolh
er P ou S": GO TO 2615
2620 LET z=2: IF k$="p" THEN LET
z=3
2625 GO SUB 5000: LET c=1
2630 PRINT #z;"Item ";c:"-": GO
SUB 5020: IF q$=CHR$ 0 THEN GO
TO 2645
2635 GO SUB 5025: PRINT #z;q$;r
$;s$;u$
2640 LET c=c+1: IF z=2 THEN PRIN
T z$": GO SUB 1050
2644 GO TO 2630
2645 CLOSE #in: RETURN
2800 PRINT "SELECCIONAR UM ITEM"
2805 PRINT "Pode seleccionar po
r numero(n)""ou por letras(l)"
2810 PRINT "Escolha n ou l"
2815 GO SUB 1050: IF k$<>"n" AND
k$<>"l" THEN PRINT "So n ou
l": GO TO 2815
2820 IF k$="n" THEN GO SUB 2850
2825 IF k$="l" THEN GO SUB 2890
2830 CLOSE #n: RETURN
2850 GO SUB 5000
2855 PRINT "Qual o numero do ite
m?": INPUT x: LET n=1
2860 GO SUB 5020: IF q$=CHR$ 0 T
HEN PRINT "Fim do ficheiro.": GO
TO 2880
2865 GO SUB 5025: IF n=x THEN CL
S: PRINT q$;r$;s$;u$: GO
TO 2860
2870 LET n=n+1: GO TO 2860
2880 PAUSE 50: PRINT "Carregue
numa tecla qualquer.": GO SUB 10
50
2885 RETURN
2890 CLS: GO SUB 5000
2895 PRINT "Escreva as primeira
s letras do""primeiro registo.
Nao esqueca""as maiusculas."
2900 INPUT t$: LET y=LEN t$: CLS
2905 INPUT #in;q$: IF q$=CHR$ 0
THEN PRINT "Fim do ficheiro.":
GO TO 2920
2910 INPUT #in;r$;s$;u$: IF q$(1
TO y)=t$ THEN PRINT "q$;r$;s
$;u$": IF y>1 THEN GO TO 2920
2915 GO TO 2905
2920 PAUSE 50: PRINT "Carregue n
uma tecla qualquer.": GO SUB 105
0
2930 CLOSE #4: RETURN
2955 RETURN

```

```

3000 CLS : LET t$="LAST ROUNDUP"
3005 GO SUB 1000: PRINT "Ja gu
ardou esta versao" do programa?
" E importante faze-lo para qu
e " possa usar novamente o fiche
iro
3010 PRINT "u$ : GO SUB 1050: IF
k$="s" OR k$="S" THEN GO TO 3050
3015 CLS : PRINT "Certifique-s
e de que " o microdrive esta pro
nto. " " Carregue numa tecla para
gravar" : GO SUB 1050
3020 SAVE "m";1;"microdata" LIN
E 10
3050 GO TO 9999
4100 LET j=0: GO SUB 5000: GO SU
B 5010
4105 GO SUB 5020: IF q$=CHR$ 0 T
HEN GO TO 4115
4110 LET j=j+1: GO SUB 5025: GO
SUB 5030: GO TO 4105
4115 CLOSE #in: LET str=out: LET
j=j+1: GO SUB 2205
4120 GO SUB 5070: RETURN
4200 CLS : LET t$="ALTERAR ITEM"
: GO SUB 1000
4205 GO SUB 5040: GO SUB 5050: G
O SUB 5020: GO SUB 5025
4210 PRINT q$; "r$"; s$; "u$
4215 PRINT e$(1): INPUT q$: PRIN
T e$(2): INPUT r$:
4220 PRINT e$(3): INPUT s$: PRIN
T e$(4): INPUT u$
4225 PRINT #out; q$: PRINT #out; r
$: s$; u$
4230 GO SUB 5020: IF q$=CHR$ 0 T
HEN GO TO 4240
4235 GO SUB 5025: GO SUB 5030: G
O TO 4230
4240 GO SUB 5060: GO SUB 5070
4245 RETURN
4300 CLS : LET t$="APAGAR ITEM":
GO SUB 1000
4305 PRINT " : LET t$="AGUARDE P
OR FAVOR": GO SUB 1000
4310 GO SUB 5020: GO SUB 5025
4315 GO SUB 5020: IF q$=CHR$ 0 T
HEN GO TO 4325
4320 GO SUB 5025: GO SUB 5030: G
O TO 4315
4325 GO SUB 5060: GO SUB 5070: P
RINT " : LET t$="APAGADO": GO SU
B 1000: PAUSE 50: RETURN
4400 RETURN
5000 OPEN #in;"m";1;f$: RETURN
5010 OPEN #out;"m";1;"temp": RET
URN
5020 INPUT #in;q$: RETURN
5025 INPUT #in;r$:s$:u$: RETURN
5030 PRINT #out;q$: PRINT #out;r

```

```

s$:u$: RETURN
5040 PRINT "Qual o numero do ite
m?": INPUT z: LET n=z-1: RETURN
5050 GO SUB 5000: GO SUB 5010: F
OR j=1 TO n: GO SUB 5020: GO SUB
5025
5055 GO SUB 5030: NEXT j: RETURN
5060 PRINT #out;q$: RETURN
5070 CLOSE #in: CLOSE #out: ERAS
E "m";1;f$
5075 MOVE "m";1;"temp" TO "m";1;
f$
5080 ERASE "m";1;"temp"
5085 RETURN

```

Definição de novos conjuntos de caracteres

Existe uma série de rotinas e programas que permitem definir novos caracteres, em substituição do conjunto próprio do Spectrum e do TC 2068. A sua utilidade é óbvia, sobretudo quando se pensa (uma vez mais) na questão dos acentos e cedilhas, ou ainda na necessidade de produzir, num texto, frases ou palavras (itálico, por exemplo) diferenciadas.

O programa que se apresenta a seguir possui sobre os correntes a grande vantagem de permitir guardar em simultâneo, no Spectrum, vários conjuntos pré-definidos, qualquer deles podendo ser utilizado em qualquer momento.

No TC 2068, pode definir-se o conjunto para colocação no endereço 30 720 a 31 487 (ver manual), para utilização com o comando OUT 255,2.

Eis um exemplo interessante do tipo de caracteres que podem ser obtidos com o programa:

Bytes: SET-1

Cama se pode ver pela exem-
pla, pode definir-se qualquer
tipo de caracteres, inclusive a
imitação da escrita manual.

Embora não se tenha feito
neste exemplo, é possível a to-
tal acrescentar, por exemplo,
caracteres com cedilhas e todos
os acentos necessários a escrita
correcta em língua portuguesa
(ou noutra língua qualquer).

Para tornar a escrita em
modo Spectrum normal, basta...
como se disse nas instruções...
um simples POKE que pode fazer-
a mais de uma frase, como pode
ver-se neste exemplo.

Fig. 59

Antes de mais, copie o programa em BASIC, tal como se encontra listado, após o que deve logo guardar uma cópia em cassette (microdrive ou diskette). Seguidamente introduza a instrução GOTO 5000, o que transferirá o set de caracteres do Spectrum para o endereço 64000.

Após isto, introduza a instrução SAVE "NS" LINE 7000, e em seguida SAVE "NS" CODE 64000,768 — o que permitirá guardar o programa transformado, com o set de caracteres do Spectrum já colocado no endereço 64000 e seguintes.

Pode agora começar a trabalhar, fazendo o habitual RUN.

O computador perguntar-lhe-á qual o carácter que pretende definir e, a partir daí, poderá ir definindo todos os que entender.

Terminado o trabalho, terá simplesmente de dar a instrução SAVE "NS" LINE 7000 e ainda SAVE "NS" CODE 64000,768. (Caso pretenda conservar vários tipos de sets de caracteres, aconselho-o a manter livre uma cassette exclusivamente para o efeito.)

Quando quiser introduzir qualquer dos conjuntos, basta seguir a instrução LOAD "NS".

Nota importante: se quiser utilizar este programa «misturado» com outros (o que será, certamente, habitual), não esqueça de fazer CLEAR 63999, antes de o introduzir, a fim de proteger o set.

Em qualquer momento de um programa em que queira utilizar o novo set bastará fazer POKE 23607, 60. Para utilizar o set normal do Spectrum... POKE 23607, 249.

```
20 REM © Luis de Campos 1984
30 REM * modificado por L.C. *
40 GO SUB 8000
50 LET K$=""
60 REM * inic. progr. princ.*
62 CLS
65 PRINT TAB 20;"1": PRINT TAB
20;"2531": PRINT TAB 20;"842584
21"
70 INPUT "Que character ";C$
90 LET B=64000+8*(CODE C$-32)
100 FOR X=1 TO 8
110 INPUT "Linha ";(X);(C$);D$
120 POKE B,VAL D$
130 LET F=PEEK B
140 PRINT TAB 1;"Linha ";X;" ";
C$;" E ";D$;TAB 19;" ";GO SUB
6300
150 LET B=B+1
160 NEXT X
164 INPUT "QUER IMPRIMIR COPIA?
(S/N) ";S$
166 IF S$="S" OR S$="s" THEN GO
SUB 8500
170 GO TO 60
4090 STOP
5000 REM *TRANSFER.CARACT.DO SPE
CTRUM para FA00*
```

```
5050 LET B=64000
5060 FOR A=15616 TO 16383: POKE
B,PEEK A
5070 LET B=B+1: NEXT A
5080 POKE 23607,249
5090 STOP
6300 REM *visao ampliada dos pix
el*
6310 FOR N=7 TO 0 STEP -1
6320 LET P=2↑N
6330 IF INT (F/P)=1 THEN PRINT "
";GO TO 6340
6335 PRINT "0";
6340 LET F=F-INT (F/P)*P
6350 NEXT N
6370 RETURN
7000 REM *chame esta linha apos
SAVE*
7010 BRIGHT 1: PAPER 7: BORDER 7
:CLS: CLEAR 63999
7020 LOAD "m";1;N$CODE
7030 POKE 23607,249: CLS
8000 REM * nome set *
8010 INPUT "Qual o nome do set "
;N$
8020 RETURN
8500 REM * copiar para printer *
8510 COPY
8520 RETURN
```

Ordenamento alfabético

Em muitos dos programas, sobretudo utilitários, de computador, surge a necessidade de proceder ao ordenamento alfabético de, por exemplo, uma lista de nomes ou fichas. As rotinas (ou programas que se encontram habitualmente em revistas ou livros, para o efeito), têm o defeito de ser extremamente morosas, normalmente porque se baseiam em princípios comparativos de sequências de todos os elementos.

O programa que a seguir é listado tem a vantagem de proceder a uma ordenação muitíssimo mais rápida (cerca de 20 vezes), podendo assim manipular grandes quantidades de dados em tempo reduzido.

O programa pode ser misturado com outro, passando a constituir uma sub-rotina, facilmente chamada a partir do «menu».

```
5000
5002 CLS : PRINT "Nome do b
loco que quer ordenar?"
5004 INPUT f$
5010 CLS : PRINT "INTRODUZA A CA
SSETTE";" NO GRAVADOR"
5100 LOAD f$ DATA w$()
5120 FOR a=1 TO 500
5130 IF w$(a, TO 4)="" " THEN
LET a=a-1: GO TO 5200
5140 NEXT a
5150
```



```

5160
5170
5180
5190
5210 LET D=1
5214 LET D=D*2
5218 IF D<=a THEN GO TO 5214
5222 LET D=INT ((D-1)/2)
5226 IF D=0 THEN GO TO 5320
5230 FOR I=1 TO a-D
5234 LET J=I
5238 LET L=J+D
5242 IF CODE W$(L, TO 1) < CODE W$(
(J, TO 1) THEN GO TO 5250
5246 GO TO 5270
5250 IF W$(L, TO 4)="" THEN
GO TO 5270
5252 LET T=W$(J)
5254 LET W$(J)=W$(L)
5258 LET W$(L)=T
5262 LET J=J-D
5266 IF J>0 THEN GO TO 5238
5270 NEXT I
5274 GO TO 5222
5320 CLS: PRINT "ORDENAMENTO
TERMINADO"
5330 INPUT "QUER GRAVAR O BLOCO?
(S/N)"; I$
5340 IF I$="N" THEN GO TO 5000
5350 CLS: PRINT "INTRODUZA A
CASSETTE" NO GRAVADOR"
5360 INPUT "NOME"; F$
5370 SAVE F$ DATA W$()
5000

```

Um copiador

Várias são as razões que tornam *indispensável*, a quem utiliza o computador para trabalho de programação, a posse de um bom copiador. Para que serve tal programa? Pois para, uma vez terminado um programa ou uma rotina, se poderem fazer cópias de segurança, por exemplo, sem necessidade de operações complicadas. Aliás, constitui um dos mais importantes princípios da programação *guardar sempre um duplicado dos programas e arquivos de dados*, uma vez que não é raro haver deterioração de uma cassette, perdendo-se assim, caso seja única, horas e horas de trabalho.

Um reparo, apenas. É perfeitamente lícito fazer cópias de programas adquiridos, com o objectivo acima indicado. Todavia, fazer tais cópias para comercialização... é *pirataria* pura. Além de pouco ética, é de resto ilegal.

O copiador cuja listagem se segue, foi adaptado de um programa da revista espanhola «Microhoby», e é constituído por duas partes:

1.ª Um curto programa em BASIC, que irá servir para carregar a parte em código máquina.

```

10 BORDER 0: INK 0: PAPER 0: F
LASH 0: BRIGHT 0: OVER 0: INVERS
E 0
20 LET TOP=PEEK 23732+256*PEEK
23733
30 CLEAR TOP
40 LOAD "copi" CODE 16384
50 RANDOMIZE USR 17010

```

Após ser teclado, deve ser gravado em cassette, com a seguinte instrução:

SAVE "MICROCOPI" LINE 10

2.ª Um extenso programa (que deve guardar em cassette de segurança, após realizado o trabalho de que adiante falaremos), que serve apenas para produzir a parte máquina do copiador.

Tecla *cuidadosamente* este segundo programa, sobretudo as linhas DATA. Grave-o em cassette de segurança. Uma vez feito isso, introduza RUN e aguarde até receber ordem para gravar o código máquina.

Na cassette reservada ao copiador (uma vez pronto) ficará, pois, apenas o primeiro programa, seguido do código máquina produzido pelo segundo, sendo a listagem deste a seguinte:

```

1 REM ***Copiador***
10 DATA "00000000000000000000"
,0
,34 DATA "0000000000000000160C"
,30 DATA "011101100712004C4F4E"
,293
,40 DATA "47495455443A20FF160E"
,752
,50 DATA "01110110071200434F4D"
,283
,60 DATA "49454E5A4F3A20FF1612"
,774
,70 DATA "01110210071200206C3D"
,262
,80 DATA "6C6F6164202020202073"
,691
,90 DATA "3D736176652020202020"
,652
,100 DATA "463D66696E2020FF1612"
,807
,110 DATA "01110210071201434152"
,276
,120 DATA "47414E444F1200202042"
,509
,130 DATA "7265616B207061726120"
,903
,140 DATA "7061726172FF16120111"
,847
,150 DATA "02100712014752414241"
,393
,160 DATA "4E444F12002020427265"
,588

```

```

170 DATA "616B2070617261207061"
897
180 DATA "726172FF161201110010"
654
190 DATA "00120020202020202020"
242
200 DATA "20202020202020202020"
320
210 DATA "20202020202020202020"
320
220 DATA "20202020FF1612011102"
443
230 DATA "100712014552524F5220"
468
240 DATA "44452043415247411200"
537
250 DATA "202070756C736520456E"
828
260 DATA "746572FF161201110210"
1052
270 DATA "071201434F5252454354"
555
280 DATA "4F1200202070756C7365"
714
290 DATA "20456E746572FF161201"
838
300 DATA "1102100712014E4F2043"
317
310 DATA "4142451200202070756C"
619
320 DATA "736520456E746572FF06"
1019
330 DATA "0FCD440EC9626B010000"
709
340 DATA "03233EFFBE20F9CD3C20"
1123
350 DATA "C9FB213B5CCB6E28F8C9"
1438
360 DATA "FBAF213B5CCB6E28F73A"
1268
370 DATA "085CCBAEFE0D20EEC900"
1215
380 DATA "00000000000000000000"
0
390 DATA "00000000000000000000"
0
400 DATA "00000000000000000000"
0
410 DATA "0000000000000000007F00"
127
420 DATA "00000000000000000000"
0
430 DATA "00000000000000000000"
0
440 DATA "001B0000000000000000"
27
450 DATA "00000000000000000000"
0
460 DATA "00000000000000000000"
0

```

```

470 DATA "0000000000007F7F7F7F"
508
480 DATA "7F007F007F7F007F7F00"
762
490 DATA "7F7F7F002424002D2D2D"
588
500 DATA "00363636001B00007F00"
316
510 DATA "7F007F007F007F00007F"
635
520 DATA "00007F007F002400002D"
335
530 DATA "002D00360036001B0000"
180
540 DATA "7F0000007F007F007F7F"
635
550 DATA "007F00007F7F7F002424"
550
560 DATA "002D2D2D00363636001B"
324
570 DATA "00000000000000000000"
0
580 DATA "00000000000000000000"
0
590 DATA "0000000000000000360000"
54
600 DATA "00000000000000000000"
0
610 DATA "00000000000000000000"
0
620 DATA "00000000000000000000"
0
630 DATA "0000000000001AF216A5C"
407
640 DATA "772AB25C2B2B36422B36"
734
650 DATA "9B3E02CD0116AF321140"
753
660 DATA "1171412100581AFE01CA"
799
670 DATA "9E4277231318F5CD3F41"
999
680 DATA "113A40CD4541FBAF213B"
996
690 DATA "5CCB6E28F73A085CCBAE"
1227
700 DATA "FE46CC0000FE6CCA0343"
1152
710 DATA "FE7320E43A1140FE0028"
1052
720 DATA "DD11AE40CD4541118840"
1032
730 DATA "CD4541DD210040111100"
591
740 DATA "AFCD0C2040632FB7610FD"
1272
750 DATA "3EFFED5B0B40DD2A535C"
1158
760 DATA "CDC20411AE40CD454111"
1014

```

```

770 DATA "FE40CD4541CD5541C39E"
1365
780 DATA "42CD3F41116240CD4541"
917
790 DATA "AF321140111100DD2100"
594
800 DATA "4037CD560530E63A0040"
815
810 DATA "FE0430DF3E16D73E09D7"
1114
820 DATA "3E01D73A004011C009CD"
823
830 DATA "0A0CC33C433E3FC9060A"
825
840 DATA "110040131AFE20DC3943"
756
850 DATA "FE80D43943D710F11112"
1225
860 DATA "40CD4541ED4B0B40CD2B"
1038
870 DATA "2DCDE32D3A0040FE03CA"
1103
880 DATA "7A43FE00C28A43ED4B0D"
1157
890 DATA "40210F27A7ED42DA8A43"
1044
900 DATA "112640CD4541ED4B0D40"
847
910 DATA "CD2B2DCDE32D2AB25C01"
1083
920 DATA "2800A7ED42ED4B535CA7"
1164
930 DATA "ED42ED4B0B40A7ED42D2"
1370
940 DATA "B64311AE40CD4541111F"
891
950 DATA "41CD4541CD5E41C39B42"
1184
960 DATA "3EFFDD2A535CED5B0B40"
1158
970 DATA "37CD5605DAD94311AE40"
1108
980 DATA "CD454111D740CD4541CD"
1179
990 DATA "5E41C39B4211AE40CD45"
1104
1000 DATA "4111FE40CD45413E0132"
852
1010 DATA "1140CD5E41C39E420000"
864
9000 CLEAR 30999: RESTORE
9010 LET A=10: LET B=11: LET C=1
9020 LET D=13: LET E=14: LET F=15
9030 FOR n=10 TO 1010 STEP 10
9040 READ h$
9050 LET cont=0
9060 FOR m=1 TO 19 STEP 2
9070 LET byte=16*VAL h$(m)+VAL h$(m+1)

```

```

9080 LET cont=cont+byte
9090 POKE end,byte
9100 LET end=end+1
9110 NEXT m
9120 READ ctrl
9130 IF cont<>ctrl THEN PRINT "E
rro na linha";n: STOP
9140 NEXT n
9150 PRINT "Correcto. Grave o co
digo maquina"
9160 SAVE "copi"CODE 31000,1010
9170 PRINT "Volte com a fita a
tras para verificar"
9180 VERIFY "copi"CODE 31000,101
0
9190 CLS: PRINT ""A gravacao
esta O.K."
9200 STOP

```

O programa, tal como se encontra elaborado, começa a funcionar automaticamente, uma vez carregado, não oferecendo qualquer dificuldade.

No menu há apenas três opções: load, save e fim. Quando se carrega um programa que se pretende copiar, o menu dá indicações sobre o tipo de programa (Basic, Máquina, Array, etc.); o seu comprimento; e a linha em que começa, ou o endereço inicial, no caso de ser em código máquina.

Scroll lateral

Esta curta rotina permite-lhe deslocar lateralmente no écran uma palavra ou frase.

```

10 RESTORE: FOR I=40000 TO 40
018: READ X: POKE I,X: NEXT I
20 DATA 33,0,64,14,192,6,32,18
3,203,30,203,30,35,16,249,13,32,
243,201
30 LIST
40 FOR I=1 TO 40: RANDOMIZE US
R 40000: NEXT I

```

Apagar palavras no écran

Por vezes necessitamos apagar palavras já impressas no écran, sem interromper o programa. Aqui está uma rotina curta, que produz esse efeito.

```

10 REM apagar
20 DIM b$(32): LET x=0: LET y=
0
30 LET a$="microcomputador": P
RINT AT x,y;a$
40 PAUSE 100: GO SUB 9000

```

```

50 STOP
9000 PRINT AT x,y;b$( TO LEN a$)
: RETURN

```

Inputs em qualquer ponto do écran

É muito fácil introduzir num programa duas linhas que permitam fazer inputs em qualquer ponto do écran, ao invés da habitual limitação das duas últimas linhas. São apenas...

```

10 INPUT AT 22,0; AT 0,0; "nome", n$
20 PRINT AT 0,0; "nome", n$

```

Variáveis em INPUT

É também muito fácil programar INPUTS onde seja possível introduzir variáveis. Exemplo:

```

10 LET A$= "programa"
20 INPUT "nome do " — A$; B'

```

Inversão do écran

Rotina que permite observar o écran como se nos encontrássemos no interior do televisor, portanto com tudo o que ali se encontra invertido. Pode, como qualquer das outras, ser introduzida num outro programa.

```

100 FOR n=50000 TO 50053: READ
a: POKE n,a: NEXT n
110 RANDOMIZE USR 50000
120 DATA 33,0,64,17,31,64,6,192
,197,213,229,6,15,126,205,122,19
,79,26,205,122,195,119,121,18,3
,27,16,240,225,1,32,0,9,209,235
,9,235,193,16,223,201,197,6,6,79
,203,41,203,23,16,250,193,201

```

Catalogador

Embora curto e simples — como de resto tem sido nossa preocupação genérica —, o programa que se segue será, talvez, um dos mais úteis para trabalho com o computador. Trata-se de um catalogador. Introduzindo o programa, o mesmo fará o catálogo completo de qualquer cassette, com indicação dos programas, seu tipo, linha de início e comprimento, podendo imprimir em écran ou em impressora:

```

10 REM catalogador
30 CLEAR 29999: BORDER 0: PAPE
R 0: INK 7: CLS
40 LET cont=1
50 RESTORE : GO SUB 350
60 INPUT "Nome da cassette:";
LINE t$: IF LEN t$>15 THEN LET t
$=t$( TO 15)
70 FOR n=1 TO (15-LEN t$): LET
t$=t$+CHR$ 32: NEXT n
80 INPUT "Lado:"; LINE f$: IF
LEN f$>1 THEN LET f$=f$(1)
90 IF f$="" THEN LET f$=CHR$ 3
2
100 LET d$=CHR$ 138+"Cass.:"+CH
R$ 32+t$+CHR$ 32+"Lado:"+CHR$ 32
+f$+CHR$ 133
110 LET i$=""
120 LET d$=i$+d$
130 LET i$=""
140 LET s$=CHR$ 138+"N. Tipo No
me Início Comp."+CHR$ 133
150 LET d$=d$+i$+s$+i$
160 FOR n=2 TO 3
170 PRINT #n;d$
180 NEXT n
190 LET a$=""
200 LET j$=STR$ cont: IF LEN j$
<2 THEN LET j$=CHR$ 32+j$
210 RANDOMIZE USR 30000
220 FOR n=30021 TO 30030
230 IF PEEK n>31 AND PEEK n<144
THEN LET a$=a$+CHR$ PEEK n
240 IF PEEK n<32 OR PEEK n>143
THEN LET a$=a$+CHR$ 63
250 NEXT n: LET tipo=PEEK 30020
260 GO SUB 360
270 LET w$=j$+CHR$ 32+t$+CHR$ 3
2+a$+CHR$ 32+c$+CHR$ 32+t$
280 PRINT w$
290 LPRINT w$
300 RANDOMIZE USR 30000
310 LET cont=cont+1: IF cont>16
THEN PRINT AT 1,0;d$: RANDOMIZE
USR 3190: PRINT AT 19,0
320 IF cont>99 THEN LET cont=0
330 GO TO 190
340 DATA 62,0,221,33,66,117,17,
17,0,55,205,66,5,201
350 FOR n=30000 TO 30013: READ
x: POKE n,x: NEXT n: RETURN
360 IF tipo=0 THEN LET t$="Prog
370 IF tipo=1 THEN LET t$="DatN
380 IF tipo=2 THEN LET t$="Dat$
390 IF tipo=3 THEN LET t$="Byte
400 LET c$=STR$ (PEEK 30033+256
*PEEK 30034): LET c$="" +c$

```

INSTRUÇÕES ÚTEIS

- Esconder linhas de um programa:
POKE PEEK 23635 + 256 * PEEK 23636, n
onde n pode ter qualquer valor acima de 64
- Bloquear maiúsculas: — POKE 23658, 8
- Deslocar linhas do programa para o fim:
POKE PEEK 23635 + 256 * PEEK 23636, n
onde n pode ter um valor entre 40 e 63
- Espera de Input: — PAUSE 0
- Esconder nome do programa: — POKE 23570,0
- Linha 0:
1 — introduz-se a linha e ENTER
2 — faz-se POKE 1 + PEEK 23635 + 256 * PEEK 23636,0
- alterações do teclado:
 - POKE 23607,58 — tudo negro
 - POKE 23607,54 — ilegível
 - POKE 23607,2 — altera os comandos
 - POKE 23607,5 — torna o teclado invisível
 - POKE 23606,8 — baralha os comandos

Cass.: MICROESCOLA		Lado: A	
N.	Nome	Início	Comp.
1	Prog EXP.SECXU		17738
2	Byte expansao	56500	9038
3	Prog testeHist	9900	4527
4	Byte code	43804	21731
5	Prog ESTATIST.		4844
6	Prog frances		1348
7	Prog analifras		317
8	Prog desenho		328
9	Prog constelac		3109
10	Prog climas		2753
11	Prog mec.celest		1091
12	Prog life		152
13	Byte LIFE	26750	300
14	Prog musica		1672
15	Prog escrevmusn		589
16	Prog piano		2938

Fig. 59-B

- sonorização do teclado:
POKE 23609, n
onde n pode adquirir qualquer valor positivo
- interrogação automática nos INPUTS:
POKE 23617,236
- desaparecimento do programa:
POKE 23755,100
- reaparecimento:
POKE 23755,0
- scroll contínuo:
POKE 23692,0
- impressão nas duas últimas linhas do écran
POKE 23659,1
- inicializar variáveis a zero (0):

Por vezes, e após testar rotinas num programa, necessitamos gravá-lo, mas «puro», isto é, sem espaço de memória ocupado com instruções DIM, por exemplo. Para tanto, basta introduzir CLEAR e ENTER, antes de fazer o SAVE.

FICHEIRO DE ALUNOS

Já atrás se deu uma listagem sobre organização de uma base de dados. Listagem pequena e limitada, que no entanto servirá para aqueles que têm ainda pouca prática ou outro tipo de limitações.

Todavia, entendi dever fornecer um programa que permite a organização, muito completa, de um ficheiro de alunos — necessidade sentida por qualquer professor (ou até pela secretaria da Escola). De qualquer modo, vai listado na sua versão mais simples — em BASIC e para cassette —, podendo ser facilmente transformado para microdrive ou diskette, o que o tornará praticamente ilimitado e de acesso muito mais rápido.

Tal como se encontra, o programa permite:

- abrir um novo ficheiro (ou vários);
- abrir, apagar ou alterar qualquer ficha, aleatoriamente;
- fazer listagens (no écran ou na impressora):
 - gerais;
 - por turma;
- fazer busca no ficheiro.

Cada ficha contém campos para:

- nome;
- número;
- ano;
- turma;
- filiação;
- morada;
- telefone;
- resultados de avaliação;
- observações.

Nota muito importante: o programa é de meu Copyright, razão pela qual pode ser utilizado por qualquer pessoa que o copie do livro ou o ceda a um colega. Todavia, *não pode ser copiado para comercialização.*

```

10 CLS
30 PRINT "***** * * FICHEIRO
DE ALUNOS * *"
40 PRINT "***** PROGRAMA DE
ACTUALIZACAO *****"
50 PRINT "*****@ Luis Campos
1985"
65 PAUSE 150
74 POKE 23509,40
80 CLS
90 PRINT "*****INTRODUZA A DAT
A DE HOJE"
92 PRINT "*****TAB 10;"COMO O EX
EMPLO";TAB 12;"12.12.84"
96 INPUT "DATA-";h$
98 IF LEN h$<>8 THEN GO TO 96
100 CLS : PRINT "*****TAB 12;
h$
102 GO SUB 1000
104 IF i$="n" OR i$="N" THEN GO
TO 80
150 GO SUB 4800
200 CLS
215 PRINT TAB 10;"MENU PRINCIPA
L"
220 PRINT "*****1 - ABRIR NOVO F
ICHEIRO"
230 PRINT "*****2 - ACTUALIZAR FI
CHEIRO"
240 PRINT "*****3 - LISTAGEM ACTU
AL"
245 PRINT "*****4 - COPIA DE SEGU
RANCA"
260 INPUT a
262 IF a=2 THEN GO SUB 3700: GO
SUB 8100
270 IF a=3 THEN GO TO 2000
275 IF a=4 THEN GO TO 5000
290 IF a>4 OR a<1 THEN GO TO 26
0

```

```

400 CLS : PRINT "*****ACTUALIZACAO
DO FICHEIRO*****"
410 PRINT "*****1 - ABERTURA DE
FICHA"
420 PRINT "*****2 - ALTERACAO D
E FICHA"
422 PRINT "*****3 - ANULACAO DE
FICHA"
430 INPUT p
438 IF p=3 THEN GO TO 1500
440 IF p=2 THEN GO TO 600
444 LET d=j
450 CLS : PRINT "*****ABERTURA
DE FICHA*****"
452 LET d=d+1
454 LET c=c+1
456 LET d$=STR$ d
458 PRINT AT 2,0;"FICHA N.-";d
460 PRINT AT 21,0;"MAXIMO
DE 25 CARACTERES";FLASH 1;" ";F
LASH 0
475 PRINT AT 6,15;"ANO:";s$: GO
SUB 1000
477 IF i$="n" OR i$="N" THEN GO
TO 475
478 INPUT "TURMA:";x$
479 IF LEN x$>2 THEN GO TO 544
480 PRINT AT 6,23;"Turma:";x$:
GO SUB 1000
481 IF i$="n" OR i$="N" THEN GO
TO 478
482
483 PRINT AT 20,0;"MORADA(rua/A
ven./Pr.,N.,etc.)";"MAXIMO DE 25
CARACTERES";FLASH 1;" ";FLA
SH 0;" "
484 INPUT m$
486 IF LEN m$>25 THEN GO TO 484
488 PRINT AT 8,0;"MORADA:";m$
490 GO SUB 1000
494 IF i$="n" OR i$="N" THEN GO
TO 484
496 PRINT AT 20,0;"MORADA(Cod.P
ost.,Localidade)";"MAXIMO DE 15 C
A";FLASH 1;"R";FLASH 0;"ACTERE
S"
498 INPUT t$
500 IF LEN t$>15 THEN GO TO 498
510 PRINT AT 10,7;t$
520 GO SUB 1000
528 IF i$="n" OR i$="N" THEN GO
TO 498
530 PRINT AT 20,0;"
532 PRINT AT 21,0;"MAXIMO DE 7
CARA";FLASH 1;"C";FLASH 0;"TER
ES"
534 INPUT "TELEFONE-";p$
536 IF LEN p$>7 THEN GO TO 534
538 PRINT AT 12,0;"TELEF.-";p$

```

```

462 INPUT "NOME: "; n$
464 IF LEN n$ > 22 THEN GO TO 462
466 PRINT AT 4,0;"NOME: "; n$; GO
SUB 1000
468 IF i$="n" OR i$="N" THEN GO
TO 462
470 INPUT "NUMERO: "; r$
472 PRINT AT 4,28;"N."; r$; GO S
UB 1000
474 IF i$="n" OR i$="N" THEN GO
TO 470
475 INPUT "ANO: "; s$
480 GO SUB 1000
542 IF i$="n" OR i$="N" THEN GO
TO 534
543 PRINT AT 21,0;"
544 PRINT AT 21,0;"MAXIMO DE 25
CARACTERES
; FLASH 0
545 INPUT "Obs. "; o$
546 IF LEN o$ > 25 THEN GO TO 545
547 PRINT AT 14,0;"Obs. "; o$; G
O SUB 1000
548 IF i$="n" OR i$="N" THEN GO
TO 545
549 PRINT AT 20,0;"

550
554 GO SUB 1200
568 LET c$(c)=n$+m$+t$+p$+x$+o$
+r$+s$
572 INPUT "Quer abrir outra fic
ha?"; i$
574 IF i$="s" OR i$="S" THEN GO
TO 450
575 GO SUB 7100
576 GO TO 200
580
590
600 CLS : PRINT "ALTERACAO
DE FICHA"
610 INPUT "QUAL O CODIGO DA FIC
HA?"; z
618
620 PRINT AT 3,10;"FICHA N.- ";
z
621 LET x=z
622
630 LET n$=c$(x,1 TO 22): LET m
$=c$(x,23 TO 47): LET t$=c$(x,48
TO 62): LET p$=c$(x,63 TO 69):
LET x$=c$(x,70 TO 71): LET o$=c$
(x,72 TO 96)
632 LET r$=c$(x,97 TO 99): LET
s$=c$(x,100 TO 101)
640 PRINT AT 3,0;"1-"; n$; "2-
"; m$; "3-"; t$; "4-"; p$; "5
-"; x$; "6-"; o$

```

```

642 PRINT AT 10,0;"7-"; r$; "8
-"; s$
644 INPUT "QUAL A LINHA A ALTER
AR?"; b$: LET b=VAL b$
650 IF b=1 THEN INPUT n$
652 IF b=2 THEN INPUT m$
654 IF b=3 THEN INPUT t$
656 IF b=4 THEN INPUT p$
658 IF b=5 THEN INPUT x$
662 IF b=6 THEN INPUT o$
664 IF b=7 THEN INPUT r$
666 IF b=8 THEN INPUT s$
668 GO SUB 1200
670 LET c$(x)=n$+m$+t$+p$+x$+o$
+r$+s$
672 CLS : PRINT "FICHA
ALTERADA"
674 PRINT AT 3,10;"FICHA N.- ";
z
675 PRINT AT 3,0;"1-"; n$; "2-
"; m$; "3-"; t$; "4-"; p$; "5
-"; x$; "6-"; o$
677 PRINT "7-"; r$; "8-"; s$
678 GO SUB 1000
680 IF i$="n" OR i$="N" THEN CL
S : GO TO 640
682 INPUT "QUER ALTERAR OUTRA L
INHA?"; i$
686 IF i$="s" OR i$="S" THEN CL
S : GO TO 640
692 INPUT "QUER ALTERAR MAIS FI
CHAS?"; i$
693 IF i$="n" OR i$="N" THEN GO
SUB 7100: GO TO 200
696 IF i$="s" OR i$="S" THEN IN
PUT "Qual o numero?"; z: GO TO 62
0
700 GO TO 200
1000 INPUT "Esta correcto? (s/n)
"; i$
1010 RETURN
1200
1202 IF LEN r$=3 THEN GO TO 1208
1204 IF LEN r$=2 THEN LET r$="0"
+r$: GO TO 1208
1206 IF LEN r$=1 THEN LET r$="00
"+r$: GO TO 1208
1208 IF LEN n$=22 THEN GO TO 127
0
1210 LET z$=""
1220 LET y=25-LEN n$
1230 FOR k=1 TO y
1240 LET z$=z$+e$
1250 NEXT k
1260 LET n$=n$+z$
1270 IF LEN m$=25 THEN GO TO 130
0
1280 LET z$=""
1290 LET y=25-LEN m$

```

```

1292 FOR k=1 TO y
1294 LET z$=z$+e$
1296 NEXT k
1298 LET m$=m$+z$
1300 IF LEN t$=15 THEN GO TO 134
0
1310 LET z$=""
1315 LET y=15-LEN t$
1320 FOR k=1 TO y
1325 LET z$=z$+e$
1330 NEXT k
1335 LET t$=t$+z$
1340 IF LEN p$=7 THEN GO TO 1380
1350 LET z$=""
1355 LET y=7-LEN p$
1360 FOR k=1 TO y
1365 LET z$=z$+e$
1370 NEXT k
1375 LET p$=p$+z$
1380 IF LEN o$=25 THEN GO TO 140
0
1382 LET z$=""
1384 LET y=25-LEN o$
1386 FOR k=1 TO y
1388 LET z$=z$+e$
1390 NEXT k
1392 LET o$=o$+z$
1400 IF LEN s$<2 THEN LET s$="0"
+s$
1402 IF LEN x$<2 THEN LET x$="0"
+x$
1410 RETURN
1500 CLS : PRINT "ANULADA"
02 FICHA
1510 INPUT "QUAL O CODIGO DA FICHA?";z
1512 PRINT "FICHA N. - ";z
1522 LET x=z
1525 LET n$=c$(x,1 TO 22): LET m$=c$(x,23 TO 47): LET t$=c$(x,48 TO 62): LET p$=c$(x,63 TO 69): LET o$=c$(x,70 TO 71): LET s$=c$(x,72 TO 96)
1527 LET c$=c$(x,97 TO 99): LET s$=c$(x,100 TO 101)
1528 PRINT AT 3,0;"1-"n$;"2-"m$;"3-"t$;"4-"p$;"5-"o$;"6-"s$
1529 PRINT AT 10,0;"7-"c$;"8-"s$
1530 INPUT "E ESTA A FICHA?";i$
1532 IF i$="n" OR i$="N" THEN GO TO 1550
1534 LET c$(x)=" "
1536 CLS
1538 PRINT "FICHA ";c$(x, TO 3);" ANULADA"
1540 INPUT "Quer anular outra?";i$

```

```

1542 IF i$="n" OR i$="N" THEN GO TO 1550
1546 IF i$="s" OR i$="S" THEN IN PUT "Qual o numero?";z: CLS : GO TO 1522
1548 GO TO 1510
1558 GO SUB 1560: GO TO 200
1560
1570
1578 RETURN
1580 GO TO 200
2000 CLS
2008 LET z$="LISTAGEM DO FICHEIRO"
2010 PRINT z$
2012 PRINT "1 - VERIFICACAO"
2014 PRINT "2 - IMPRESSAO GE RAL"
2015 PRINT "3 - IMPRESSAO PO R TURMA"
2016 INPUT i
2018 IF i=2 THEN GO TO 2200
2019 IF i=3 THEN GO TO 2300
2021 CLS
2022 PRINT z$
2024 LET c=0: LET lop=3
2026 LET w$=" N. ALUNO"
2028 PRINT w$
2030 LET j$="=====
=====
2032 PRINT j$
2034
2036 FOR a=1 TO 300
2040 PRINT c$(a,97 TO 99);"- ";c$(a, TO 22);TAB 29;c$(a,100 TO 101);TAB 5;c$(a,23 TO 47);TAB 29;c$(a,70 TO 71);TAB 5;c$(a,48 TO 62);TAB 5;c$(a,63 TO 69);TAB 5;c$(a,72 TO 96)
2041 PRINT "
-----
2042 LET c=c+1
2044 IF c=lop THEN INPUT "ENTER cont. / -i- interrompe";i$: GO SUB 9000: CLS : PRINT w$;j$
2048 NEXT a
2050
2052
2054
2075 PRINT AT 20,0;"FIM DO FICHEIRO"
2078 INPUT "Carregue ENTER para Menu.";i$
2080 GO TO 200
2100
2120
2128

```



```

2130
2132
2134
2136 LOAD *"m";2;z$ DATA c$( )
2138 FOR a=1 TO 100
2140 PRINT c$(a, TO 3);"- ";c$(a
;4 TO 25);";TAB 5;c$(a,26 TO 50)
;";TAB 5;c$(a,51 TO 65);";TAB 5;
c$(a,66 TO 72);TAB 30;c$(a,73);
;TAB 5;c$(a,74 TO )
2141 PRINT "-----"
"
2142 LET c=c+1
2144 IF c=lop THEN INPUT "ENTER
cont. / -i- interrompe";i$; GO S
UB 9000: CLS : PRINT w$;";j$
2148 NEXT a
2150 PRINT AT 20,0;"FIM DO BLOCO
"
2152 INPUT "Quer listar outro bl
oco?";i$
2154 IF i$="s" OR i$="S" THEN LE
T i=2: CLS : GO TO 2120
2156 GO TO 200
2158
2200 CLS : PRINT z$
2202 LET z$="LISTAGEM DO FICHEIR
O DE ALUNOS"
2204 LET c=1: LET lop=9: LET l=1
2206 LET f=0
2208 PRINT "-----"
2208 PRINT "NÃO ESQUEÇA LIG
AR A IMPRESSORA"
2208 PRINT "E METER PAPE
L SUFICIENTE"
2208 PRINT "QUANDO TUDO ESTIV
ER EM ORDEM"
2208 PRINT "CARREGUE NUMA TEC
LA"
2209 INPUT i$
2210 LPRINT
2211 LPRINT "-----"
"
2212 LPRINT ;TAB 10;z$;" em ";h
$
2214 LPRINT ;TAB 5;"-----"
"
2220
2222 GO SUB 2234
2224
2232 GO TO 200
2234 FOR c=1 TO 300
2236 LET a$=STR$ c: GO SUB 2252
2238 LPRINT ;TAB 5;c$(c, TO 3);
"- ";c$(c,4 TO 25);";TAB 10;c$(c
,26 TO 50);";TAB 10;c$(c,51 TO 6
5);";TAB 10;c$(c,66 TO 72);";TAB
10;c$(c,73);";TAB 10;c$(c,74 TO
)
2238 LET f=f+1

```

```

2240 IF f=lop THEN LPRINT " ";"-
-----"
"
2242 GO SUB 9100: LPR
INT ;TAB 70;"pag.";l;
2246 NEXT c
2248 RETURN
2252 IF c<10 THEN LET a$="00"+ST
R$ c: RETURN
2256 IF c>9 AND c<100 THEN LET a
$="0"+STR$ c: RETURN
2258 IF c$(c, TO 3)="" THEN G
O TO 2280
2260 LET a$=STR$ c: RETURN
2262 LPRINT
2264 LET t=lop-f
2266 FOR a=1 TO t
2268 LPRINT " ",,
2270 NEXT a
2272 LPRINT " ";"-
-----"
"
2290 GO TO 200
2300 CLS : PRINT "-----"
2300 CLS : PRINT "-----"
2302 PRINT "1 - UMA TURMA
RA"
2304 INPUT i$
2306 LET i=VAL i$
2308 IF i=1 THEN GO TO 2500
2310 LET z$="LISTAGEM DO FICHEIR
O DE ALUNOS - POR TURMA"
2312 LET l=1: LET f=0: LET lop=1
0
2314 LPRINT
2316 LPRINT "-----"
"
2316 LPRINT ;TAB 10;z$;" em ";h$
2318 LPRINT ;TAB 5;"-----"
"
2330 FOR e=1 TO 8
2332 LPRINT ;TAB 30;"TURMA - ";e
2340 GO SUB 2354
2342
2344
2352 NEXT e
2354 FOR c=1 TO 300
2356 LET a$=STR$ c: GO SUB 2380
2360 IF c$(c,73)=STR$ e THEN LPR
INT ;TAB 5;c$(c, TO 3);"- ";c$(
c,4 TO 25);";TAB 10;c$(c,26 TO 5
0);";TAB 10;c$(c,51 TO 65);";TAB
10;c$(c,66 TO 72);";TAB 10;c$(c
,74 TO ); LET f=f+1
2366 IF f=lop THEN LPRINT " ";"-
-----"

```

```

-----": GO SUB 9200: LPR
INT "TAB 70;" pag.": l;
2368 NEXT c
2370 RETURN
2380 IF c<10 THEN LET a$="00"+ST
R$ c: RETURN
2382 IF c>9 AND c<100 THEN LET a
$="0"+STR$ c: RETURN
2384 IF c$(c, TO 3)=" " THEN G
O TO 2388
2388 LET a$=STR$ c: RETURN
2389 LET t=lop-f
2390 LPRINT
2391 FOR a=1 TO t,
2392 LPRINT
2394 NEXT a
2396 LPRINT "":
-----"
2398 IF e<8 THEN GO TO 2352
2400 GO TO 200
2500 CLS: INPUT "Qual a turma?"
e
2504 LET z$="LISTAGEM DO FICHEIR
O DE ALUNOS DA TURMA - "
2506 LET l=1: LET f=0: LET lop=1
0
2507 LPRINT
2508 LPRINT "-----"
-----"
2510 LPRINT "TAB 10;z$;e;" em
" h$
2512 LPRINT "TAB 5;"
-----"
2514 GO SUB 2526
2516
2518
2520 FOR c=1 TO 300
2522 LET a$=STR$ c: GO SUB 2538
2530 IF c$(c,73)=STR$ e THEN LPR
INT "TAB 5;c$(c, TO 3);"-":c$(
c,4 TO 25);TAB 10;c$(c,26 TO 5
0);TAB 10;c$(c,51 TO 65);TAB
10;c$(c,66 TO 72);TAB 10;c$(c
,74 TO ): LET f=f+1
2532 IF f=lop THEN LPRINT "":
-----"
-----": GO SUB 9200: LPR
INT "TAB 70;" pag.": l;
2534 NEXT c
2536 RETURN
2538 IF c<10 THEN LET a$="00"+ST
R$ c: RETURN
2540 IF c>9 AND c<100 THEN LET a
$="0"+STR$ c: RETURN
2542 IF c$(c, TO 3)=" " THEN G
O TO 2546

```

```

2544 LET a$=STR$ c: RETURN
2546 LET t=lop-f
2547 LPRINT
2548 FOR a=1 TO t,
2550 LPRINT
2552 NEXT a
2554 LPRINT "":
-----"
2556 INPUT "Quer listar outra tu
rma?" i$
2558 IF i$="s" OR i$="S" THEN GO
TO 2500
2560 GO TO 200
3000 REM avaliacao
3010
3020
3030
3040
3050
3060
3070
3700 REM ***intr.cass.dados***
3710 CLS: PRINT "Introduza no g
ravador a cassette"" com o fic
heiro de alunos."" Quando term
inar ENTER."
3720 INPUT i$
3740 LOAD f$ DATA c$()
3750 RETURN
3800
3810
3820
4800
4830 LET g$="
4900
4910 LET f$="ALUNOS"
4920 LET j=1
4925 LET a=0: LET b=0: LET m=0
4930 LET d=0
4935 LET e$=CHR$ 32
4940 DIM c$(200,98)
4990 RETURN
5000 CLS
5010
5020
7100 REM *** grav.final
7110 CLS
7120 PRINT "Coloque a cassette
de ficheiros"" de ALUNOS no gr
avador."" carregue no botao R
ECORD."" e depois em ENTER"
7130 INPUT i$
7140 SAVE f$ DATA c$()
7190 RETURN
8000
8100 REM *** busca fim ficheiro
8105 FOR c=1 TO 300
8108 IF c$(c,4 TO 6)=" " THEN

```

```

LET c=c-1: GO TO 8210
8109 LET j=c
8110 NEXT c
8202 LET j=VAL c$(c, TO 3)
8210 RETURN
8550
9000 IF i$="i" OR i$="I" THEN GO
TO 200
9010 LET lop=c+3: RETURN
9100 LET l=l+1: LET lop=f+9: RET
URN
9130 GO TO 200
9200 LET l=l+1: LET lop=f+10: RE
TURN

```

3.3 — Programas comerciais

Em fins de 1984 ultrapassava os 20 000 o número de programas comerciais para o ZX Spectrum, razão pela qual se torna difícil dar uma indicação completa sobre o assunto — tanto mais que diariamente continuam a ser produzidos novos programas.

De qualquer modo, não quero deixar de indicar alguns que me parecem especialmente úteis para o ensino.

Um destaque para o programa ASTRONOMER, que se destina ao cálculo e animação dos corpos celestes, possuindo potencialidades verdadeiramente invulgares. Permitindo indicar qualquer data e visualizar qualquer zona do céu, produz, entre outros, écrans como os que seguem:

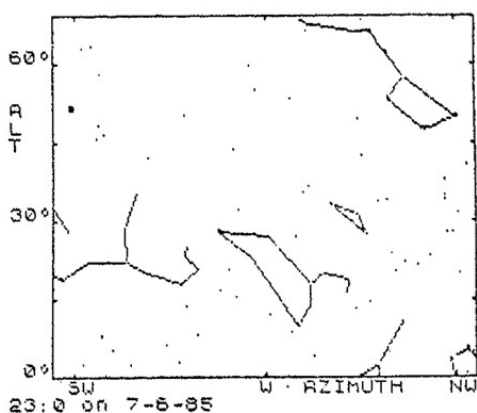


Fig. 61

MOON
 At 23h 0m 0s GMT on 7-6-1985
 Position is RA 21h 36m 33s
 Dec -20° 35' 29"
 Age since new moon is 20 days

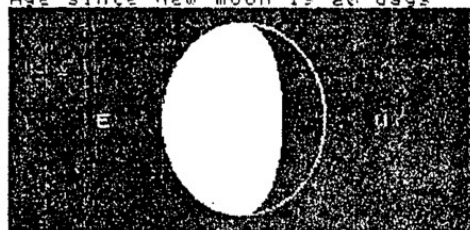


Fig. 62

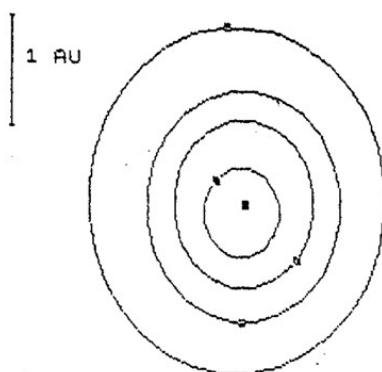


Fig 63

1985.47

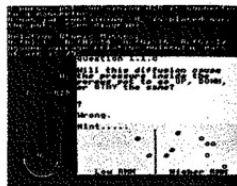
Ainda no âmbito dos programas de muito boa qualidade, são de referir seis (por enquanto), da Hill MacGibbon, especial e cuidadosamente preparados para outras tantas disciplinas do ensino secundário.

Trata-se de programas elaborados em módulos, com testes diagnóstico e um volumoso livro científico para cada disciplina. Neste momento encontram-se à venda programas de Matemática, Química, Economia, Física, Biologia e Francês.

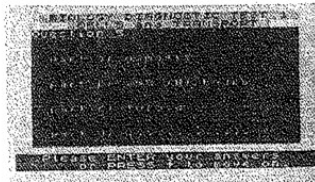
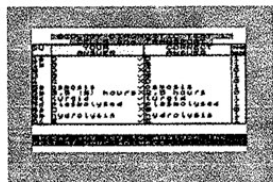
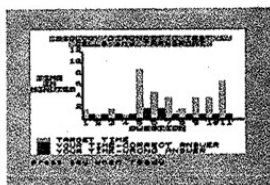
Segundo soubemos, a empresa Infornova desenvolve contactos no sentido de obter os direitos de comercialização para Portugal, tendo neste momento várias equipas de professores portugueses — ligados às diversas matérias — a realizar um trabalho de adaptação dos referidos programas.

Apenas para que possa aquilatar-se da qualidade dos mesmos, damos o exemplo dos conteúdos de alguns dos núcleos programáticos do programa de Biologia: a célula e a organização da vida; nutrição; respiração; transporte; excreção, osmoregulação e temperatura; coordenação; esqueleto e locomoção; reprodução, crescimento e desenvolvimento; genética e evolução; ecologia e microbiologia.

E estes são alguns dos ecrãs produzidos pelos programas dessa empresa, sendo de notar a perfeição e o pormenor dos gráficos, bem como a quantidade da informação fornecida em texto.



De notar que alguns dos ecrãs são dinâmicos. Os movimentos criados, aliados à correcta combinação das cores, tornam muito fácil a compreensão da matéria apresentada, bem como uma mais perfeita aquisição dos conhecimentos a que se referem.



No que respeita ao software de produção inteiramente nacional — o qual deve, em meu entender, ser apoiado e incentivado desde que possua a qualidade mínima exigível —, uma empresa se vem destacando, desde há algum tempo. Trata-se da ASTOR, que lançou no mercado, entre outros, um programa de Geografia de Portugal, que contém opções, de aprendizagem (vários níveis) de rios, elevações, localidades, etc.; e ainda testes igualmente de vários níveis. Para além disso possui uma opção de

jogo, até 6 participantes — uma espécie de sabatina. Essa empresa produziu igualmente programas de Matemática (infantil e de nível mais adiantado), obedecendo a idênticas opções — alguns até de nível universitário.

Com menor qualidade — e até com eventuais erros científicos — pode encontrar-se no mercado uma série de outros programas, desde os de línguas (alemão, inglês, francês) até outros do tipo estrutura atómica, geometria descritiva, trigonometria, programação linear, etc.

Programas utilitários

Quem pretenda trabalhar em programação tem toda a conveniência em adquirir uma série de programas chamados utilitários — aos quais de resto já me referi, contendo aliás esta obra alguns, a par de curtas rotinas e instruções.

Os programas utilitários servem, essencialmente, para fornecer rotinas especiais para a execução de curtas tarefas no interior de um programa. Ou ainda como auxiliares de conversão de códigos, etc. De entre os muitos disponíveis, há que destacar:

— *Supercode*: programa com 100 rotinas (o Supercode 3.5 possui 150), que permite, por exemplo, realizar o scroll em várias direcções e com efeitos diversos; imprimir o screen; renumerar; comprimir programas; proteger programas; copiar blocos; analisar a memória; baralhar a listagem; etc.

— *Assemblador*: permite escrever um programa em linguagem «assembler», convertendo-o depois em linguagem máquina.

— *Desassemblador*: permite realizar a operação inversa, isto é, converter código máquina (em memória) em programa assembler.

— *Compilador*: converte um programa BASIC num reduzido programa em código máquina (se bem que mais extenso que um programa máquina puro).

— *Mega Basic*: permite realizar programação em Basic mais extenso que o da Sinclair.

— *Paint Box*: o melhor programa existente para definir gráficos, áreas de um ecrã ou mesmo um ecrã inteiro.

— *Taswide*: permite utilizar o ecrã com 64 caracteres, o que o torna especialmente útil para tabelas ou texto.

— *Tasword*: programa de processamento de texto, muito completo e com acesso ao Basic — factor que permite fazer-lhe alterações, como por exemplo definir acentos e cedilhas.

Claro que se não pretende (nem pode) esgotar uma vasta listagem de programas utilitários. Muitos outros podem ser encontrados no mercado, dependendo a sua utilidade das necessidades e objectivos de trabalho de cada um.

De qualquer modo, antes de adquirir qualquer desses programas é conveniente certificar-se exactamente das suas capacidades e — muito importante — solicitar as instruções de funcionamento, sem as quais, muitas vezes, jamais será possível tirar dos mesmos todo o partido possível.

PROBLEMAS COM O SISTEMA

- 4.1 — As ligações eléctricas
- 4.2 — O TV e a sintonia
- 4.3 — Guardar e carregar dados
- 4.4 — Os periféricos
- 4.5 — Avarias diversas

4.1 — As ligações eléctricas

Por vezes — e sem que nenhuma razão lógica permita perceber porquê — nota-se no écran uma vibração que, embora raramente, chega a provocar uma paragem do programa ou até a destruir alguns dados que se encontravam nas memórias RAM. Tal facto deve-se a um de dois factores:

1 — variação brusca (quebra) de tensão da rede de distribuição eléctrica — factor frequente em certas zonas de cidade, sobretudo à noite ou em períodos de grande consumo;

2 — sobrecarga súbita de tensão, provocada por um dos aparelhos do sistema — muito frequentemente o simples acto de ligar ou desligar a impressora, por exemplo.

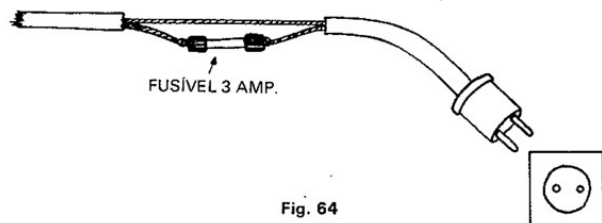


Fig. 64

Anular o segundo efeito é fácil. Como em outro capítulo se disse, o computador deve, de preferência, ser ligado a uma tomada independente relativamente aos outros aparelhos do sistema; estes (televisor, gravador,

impressora, etc.) devem ser, portanto, ligados a outra tomada. Deste modo se evitarão interferências.

Por outro lado, convém (não é indispensável) intercalar um fusível de 3 ampères entre um dos fios de corrente do computador (*não os fios que vão deste à fonte de alimentação!!!* nesses não deve mexer nunca) e a tomada da rede.

A anulação do primeiro factor é menos fácil. Todavia poderá adquirir um conjunto de alimentação constituído por bateria de cádmio, o qual será muito útil sobretudo em zonas onde há cortes frequentes de energia, uma vez que evita a perda de trabalho realizado ou em realização.

Por vezes um simples estabilizador pode solucionar o problema.

4.2 — O televisor e a sintonia

Sucede com frequência que, ao ligar o computador e o televisor, se verifica ter havido uma *estranha* perda de sintonia, tanto mais estranha quanto é sabido que ela era perfeita na última vez que utilizámos o sistema.

Não há razão para alarmes, pois isso se deve tão só à baixa qualidade do modulador que habitualmente equipa os micros, e basta dar um toque no botão de sintonia do televisor para tudo se normalizar.

De qualquer modo — e porque os manuais não são completos a este respeito — convém adiantar algumas normas para a «afinação» inicial do TV. Vamos falar apenas da sintonia de um TV a cores — pois esta é um pouco mais difícil de conseguir — e, assim, quem possuir TV a preto e branco poderá seguir as mesmas normas, excepto no que respeita à cor.

1 — através das devidas instruções, ponha o écran do TV completamente negro (cor 0) e as letras brancas. Feito isto, verifique se o negro é mesmo negro (não cinzento) e, se assim não for, utilize os botões de contraste e luminosidade do TV até obter esse efeito, devendo igualmente as letras ser perfeitamente brancas.

2 — obtido o efeito indicado, constate se há «rasto» das letras. Se o recorte das mesmas não for perfeito, rode o botão de sintonia num e noutro sentido até que isso aconteça.

3 — cumprida esta segunda fase, coloque o BORDER na cor 6, o PAPER na 2 (por exemplo) e o INK noutra. Vá agora afinar a cor. Notará, muito provavelmente, um **borrado** e interpenetração das cores (sobretudo no TC 2068) ⁽¹⁾. Rode o botão de regulação de cor do TV até que esse «borrado» desapareça ou se atenuar o suficiente — aliás o excesso de cor provoca não só esse borrado como lesões na vista, com um uso prolongado do aparelho.

4 — sintonize um segundo canal do TV, com uma ligeira diferença em relação ao que acabou de sintonizar. Por vezes, variações de corrente

⁽¹⁾ O «borrado» do TC 2068 nunca desaparecerá completamente, uma vez que se deve a um atraso de sincronismo, que a Timex não conseguiu resolver, e se prende com o facto de o computador a partir do qual este foi desenvolvido (O TS 2068, dos Estados Unidos) ter sido concebido para uma rede de alimentação com variações de ciclos diferente.

aliadas aos factores já apontados provocam uma variação de sintonia, que poderá, em alguns casos, ser contrariada pela simples mudança de canal. (No capítulo seguinte será apresentada uma solução mais dura para o caso).

4.3 — Guardar e carregar dados

Um dos problemas mais frequentes que se põe aos utentes de sistemas informáticos é a perda de informação guardada em cassette de gravador, wafer de microdrive ou diskette. A frase «o programa não entra» é comum entre utilizadores.

Sendo, é verdade, o processo de transferência de dados para os periféricos uma das operações mais delicadas do trabalho com microcomputador — logo mais sujeita a falhas — há, no entanto, algumas normas simples que permitem reduzir quase totalmente os problemas que essa transferência põe. Uns, específicos do tipo de periférico utilizado; outros, de ordem geral.

O gravador de cassettes

Trata-se do equipamento mais comum para guardar informação de computador, porque o mais barato e acessível em termos de software. Utiliza cassettes com uma fita de plástico banhada em óxido de ferro (ou outro mineral), sendo este óxido composto por biliões de pequeníssimas partículas semelhantes a agulhas magnéticas.

Em funcionamento, essa parte da fita *roça* por uma peça metálica do gravador — a cabeça —, a qual faz a «escrita» ou a «leitura» da informação contida na fita. É nesta cabeça que residem habitualmente os problemas. A sua posição, relativamente à fita, deve ser rigorosamente de um alinhamento de 90°, quer na horizontal, quer na vertical.

Em princípio, os gravadores novos — vindos de fábrica — deveriam trazer as cabeças com esse alinhamento (*azimute*) correcto. Todavia, é surpreendente o número de aparelhos deste tipo — mesmo de boas marcas e preço elevado — que trazem a cabeça desalinhada, factor que provoca reprodução incorrecta e, consequentemente, dificuldades (por vezes inultrapassáveis) na carga dos programas, sobretudo se foram gravados num aparelho com o azimute correcto.

Felizmente o problema é de fácil solução. Se bem que o ideal seja proceder a essa afinação com um osciloscópio (por um técnico), qualquer pessoa pode fazê-lo, com perfeição razoável e bastante simplicidade:

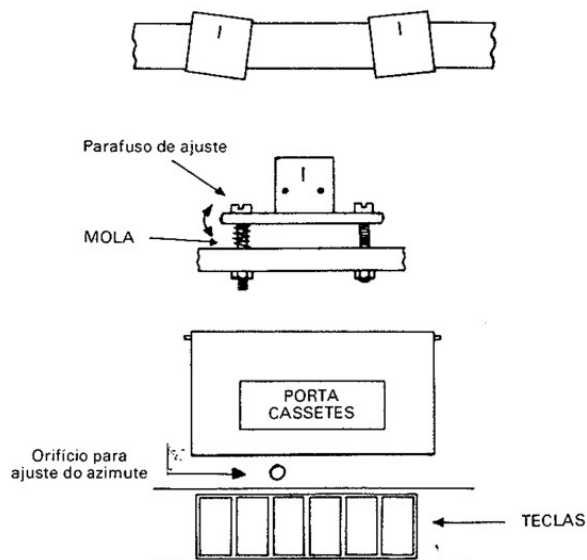
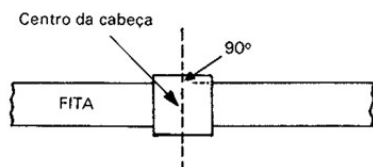


Fig. 65

- 1 — Coloque uma cassette com um programa no gravador.
- 2 — Ponha-a em movimento, ajustando o volume de som de modo a ouvir perfeitamente o sinal que for reproduzido.
- 3 — Após começar a ouvir esse sinal, introduza uma pequena chave de fendas no parafuso móvel de fixação da cabeça do gravador, rodando-o lentamente, num ou noutro sentido, até obter o som mais nítido e elevado possível.
- 4 — A operação está pronta. Em princípio, a cabeça do seu gravador está azimutada a cerca de 90° e os seus problemas desapareceram.

Um outro cuidado a ter com a cabeça do gravador é limpá-la, de tempos a tempos, com uma cottonneste embebida em álcool *etilico*, uma vez que o roçar da fita vai deixando depósito de óxido de ferro na cabeça e, ao fim de algum tempo (sobretudo se as fitas forem de má qualidade), esse depósito dificulta quer a gravação quer a audição.

O tipo de fitas que deve utilizar será a C-12, C-14, C-20 e, no máximo, a C-30.

Ao contrário dos gravadores de cassettes, as microdrives, funcionando embora segundo os mesmos princípios básicos (ex-ceto que a fita da wafer é um sem-fim), são equipamentos de alta qualidade e que, portanto, não põem os problemas de desalinhamento de cabeças.

Não se põem também problemas de sujidade de cabeças, uma vez que a fita utilizada é de óxido de crómio, de muito alta qualidade e fidelidade.

De qualquer modo, há um problema que surge, não raro: o introduzir-se uma wafer no microdrive e, após uma instrução, aparecer no écran a mensagem.

“Microdrive not present”

Dois factores podem determinar, geralmente, tal estranho acontecimento:

1 — quebra súbita de tensão, que alterou as zonas de memória de comando dos microdrives (neste caso basta desligar e ligar de novo o computador);

2 — por qualquer razão, a fita dentro da wafer deixou de estar correctamente acamada e prende (basta dar umas batidas com a wafer sobre a mesa, para a questão se normalizar).

As diskettes

Não põem qualquer problema de funcionamento.

Um problema comum

Quer trabalhe com cassettes, com wafers ou diskettes, poderá acontecer que um dia verifique, com espanto, que a informação guardada num destes suportes se deteriorou ou desapareceu. Tal facto ficou a dever-se — quase com toda a certeza — a uma desmagnetização.

É fácil acontecer que, por distração, coloque um destes suportes demasiado próximo de uma fonte magnética (são tantas, no sistema): alimentador, aparelho de TV, etc. Se isso acontecer durante um período de tempo razoável, o magnetismo será alterado e a informação destruída.

A forma de evitar isto é simples: *nunca coloque uma cassette, uma wafer ou uma diskette a uma distância inferior a 30 centímetros de uma fonte magnética*. E, sobretudo, nunca coloque nenhum destes suportes próximo da *parte de trás do televisor*.

4.4 — Periféricos

São inumeráveis os periféricos existentes para o Spectrum (e TC 2068). Por tal razão poderão ser inúmeros os problemas a surgir com os mesmos, embora tal não seja vulgar. Quero dizer, atendendo à simplici-

dade da sua arquitectura, raramente um periférico apresenta problemas de funcionamento.

Assim, apenas serão necessárias uma ou duas indicações sobre os mesmos.

Antes de mais, a regra de ouro para o trabalho com tais equipamentos — e que nunca é demais repetir, apesar de os manuais trazerem indicações sobre o assunto —: *nunca ligue ou desligue um periférico ao (ou do) computador, sem que ambos (periférico e computador) se encontrem desligados da corrente*. Se o fizer... pode simplesmente avariar completamente o periférico em causa ou «liquidar» o CPU do computador.

Um outro conselho é o de adquirir *sempre* periféricos de marca, únicos que dão garantias de funcionamento correcto e de reparação. Existem no mercado cópias «piratadas» desses periféricos — como por exemplo uma «light pen» que apareceu recentemente em lojas da especialidade. O que sucede é que tal periférico funciona mal; o seu grau de fiabilidade é baixíssimo; qualquer reparação após a garantia (que é de apenas 6 meses) torna-se cara. Mais vale, pois, gastar um pouco mais no acto da compra — até porque o comprar periféricos «piratas» é alimentar a vigarice generalizada que, infelizmente, grassa no mercado.

4.5 — Avarias diversas

O primeiro conselho que posso dar — e que servirá de regra genérica — é que, face a um qualquer disfuncionamento do computador, o entregue imediatamente a um técnico, para ser reparado.

Todavia, casos há em que é possível solucionar o problema, desde que se disponha de competência para tal e se assuma o risco daí decorrente.

É o caso, por exemplo, da avaria se manifestar por um ondular do écran, que apesar de tudo não impede o normal funcionamento do computador. Trata-se apenas de avaria no sistema de rectificação da corrente contínua, pelo que deve verificar-se o triac situado à entrada do computador.

O mesmo procedimento se deve ter no caso de se verificar que não há entrada de corrente no computador.

Se o écran ficar subitamente negro ou cheio de quadrados coloridos, sem obedecer ao teclado, é sinal certo de avaria do processador central, pelo que o mesmo terá de ser substituído.

Os problemas de sintonia ou de alterações no écran estão geralmente ligados ao modulador ou ao relógio do computador.

Para além destes casos, como disse, tratar-se-á certamente de avaria mais complexa, que só um técnico, com equipamento adequado, poderá detectar e reparar.

ALÉM DO SISTEMA

Devo começar por declinar toda a responsabilidade sobre qualquer avaria que a *má execução das instruções* que aqui indico possa provocar no sistema informático em que forem aplicadas. Para ser mais claro: *tudo* o que aqui for referido foi já aplicado e testado, sem quaisquer problemas. Todavia, nem sempre o utilizador/proprietário do computador possui a *técnica* correcta para a execução das tarefas inerentes; e é por isso que a ninguém, senão ao próprio, devem ser assacadas responsabilidades que, muito justamente, ao próprio cabem.

Por tal motivo, em caso de dúvida, o melhor será entregar tais trabalhos a um técnico.

Um outro aviso, que de resto lhe é feito no acto de aquisição de um computador: a abertura do equipamento, dentro dos prazos de garantia, faz com que essa garantia se perca — o que não significa que, em caso de avaria, o equipamento não possa ser reparado pela fábrica, claro.

Posto isto, vamos então às tais indicações que podem tornar o ZX Spectrum (ou o TC 2068) em computadores com algumas características extra, que nalguns casos poderão melhorar muito a qualidade ou facilidade do trabalho a realizar com os mesmos.

LOAD and SAVE

Não raro surgem problemas com a carga ou gravação de programas em cassette, como de resto já se indicou no capítulo anterior. E, apesar de a cabeça do gravador se encontrar já azimutada correctamente, eles por vezes persistem.

É certo que são dadas indicações no sentido de desligar a ficha EAR, quando se pretende gravar; ou a MIC, quando se pretende carregar um programa. Contudo, além de pouco prático, o estar permanentemente a executar tais operações quantas vezes não significa o perder horas de trabalho de programação, apenas porque nos esquecemos de voltar a introduzir a ficha MIC, no momento em que nos preparamos para guardar o programa em cassette.

A causa que, em alguns gravadores, dificulta por vezes este tipo de operações é o «feed-back» que se produz entre as duas fichas. Ora para

eliminar tal efeito — e nunca mais serem necessárias preocupações de desligar e voltar a ligar qualquer das referidas fichas — basta fazer o seguinte:

1.º Compre uma ficha jack de 3.5 mm, idêntica à existente no cabo de ligação do computador ao gravador.

2.º Adquirir também uma resistência (se possível miniatura) com o valor de 330 ohms.

3.º Corte o cabo referido junto à ficha EAR e solde-o na nova ficha, com a resistência de 330 ohms também soldada entre os dois polos.

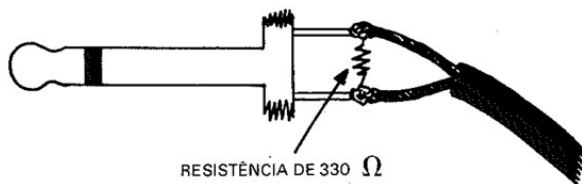


Fig. 66

Feito isto, pode efectuar as ligações, após o que desaparecerá o problema já referido, com a vantagem de passar a poder ouvir som quando o programa estiver a «entrar».

Ligar/desligar o spectrum

Outro dos «defeitos» do ZX Spectrum (já corrigido no TC 2068) é o de ter de desligar e voltar a ligar a corrente sempre que se pretenda inicializar o microprocessador ou, mais simplesmente, interromper um programa que possua uma rotina anti-Break.

É também bastante fácil solucionar este problema, embora, neste caso, tenha que *abrir* o Spectrum. Todavia, o resultado é muito compensatório.

1.º Adquirir um interruptor de pressão, idêntico ao da figura.

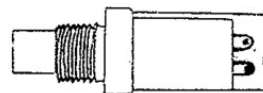


Fig. 67

2.º Solde-lhe um pedaço de fio de cobre rígido, para que possa adquirir a elevação correcta.

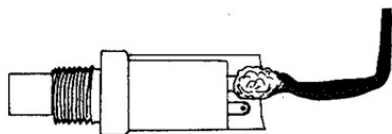


Fig. 68

3.º Faça um furo no local correcto da tampa superior de plástico do Spectrum.

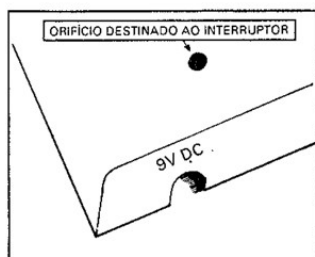


Fig. 69

4.º Solde os dois bornes do interruptor, conforme o esquema seguinte (muita atenção para não se enganar nos pontos de soldadura).

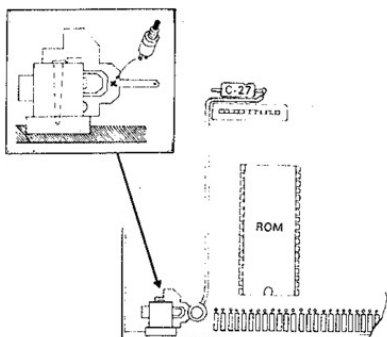


Fig. 70

ENTRADA DE 9V FICHA DE LIGAÇÃO DE INTERFACES

Após fechar novamente o computador, o trabalho está concluído. A partir de agora, sempre que queira «limpar» as memórias ou interromper um programa basta carregar no interruptor.

Sintonia e definição

Outro problema que surge com uma certa frequência — e este com maior gravidade no TC 2068 do que no Spectrum — é o da dificuldade de sintonia correcta do TV, ou, o que é pior, variações de sintonia de um dia para o outro ou até no decorrer da execução de um programa. Tal se deve ao modulador de frequência que equipa o computador.

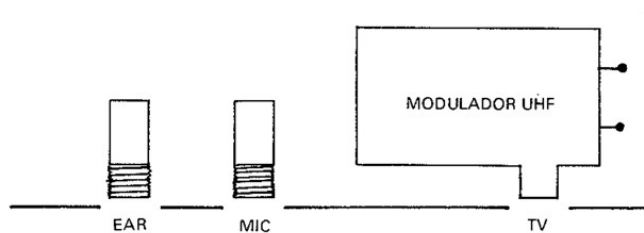


Fig. 71

Até hoje ainda não consegui perceber a que ficou a dever-se a opção que grande parte dos fabricantes de computadores tomou pelos moduladores da banda UHF. É que tal banda tem uma largura de sintonia muito crítica (estreita), o que dificulta a operação de sintonia propriamente dita, oferecendo, por outro lado, poucas garantias de manutenção da mesma — isto devido à baixa qualidade dos moduladores que habitualmente equipam os microcomputadores (mesmo os mais caros!).

Ora a solução é muito simples e eficaz.

Existem no mercado moduladores exactamente com as mesmas dimensões e características básicas — mas para VHF (banda I ou banda III) os quais têm uma largura de banda muito maior, permitindo uma sintonia fácil e estável.

Bastará, portanto, adquirir um desses moduladores (são relativamente baratos), de preferência VHF Banda I; dessoldar cuidadosamente o que se encontra instalado no computador e substituí-lo pelo novo.

A partir daí, o computador passará a funcionar na Banda I do televisor, sendo portanto necessário reservar um canal para o efeito.

III PARTE
ANEXOS

ANEXO A

Listagens de programas

*Listagem do programa de História
«A Expansão Portuguesa no séc. XV»*

```

10 POKE 23509,15
20 CLS : GO SUB 9450: REM mapa
/africa
30 GO TO 1000: REM movimento
75 GO SUB 9500: GO SUB 2000: R
ETURN
195 GO SUB 9500: GO SUB 2500: R
ETURN
240 GO SUB 9500: GO SUB 3000: R
ETURN
370 GO SUB 9500: GO SUB 4000: R
ETURN
380 GO SUB 9500: GO SUB 5000: R
ETURN
1002 FOR n=1 TO 50
1005 BRIGHT 0: READ z: READ w: R
EAD a: READ b: PRINT AT z,w: PAP
ER 5: INK 0: CHR$ 144: AT a,b: PAP
ER 5: CHR$ 128: PAUSE 25
1010 LET m$=STR$ z: LET n$=STR$
w: LET k2=VAL m$: LET k1=VAL m$+
VAL n$
1015 IF k1=7 AND k2=4 OR k1=19 A
ND k2=10 OR k1=24 AND k2=13 OR k
1=37 AND k2=20 OR k1=38 AND k2=8
THEN LET j1=k1: GO SUB j1*10
1050 NEXT n
1050 RESTORE : GO TO 1000
2000 CLS : REM bojador
2001 PAPER 0: INK 6: BRIGHT 1: R
ANDOMIZE USR 58820: PAUSE 25
2002 PRINT AT 1,1: PAPER 2: INK
7: FLASH 1: " 1415 "
2004 LET l=3: LET o$="A Expansao
portuguesa": GO SUB 7000: LET l
=l+1: LET o$="comeca com a ": GO

```

```

SUB 7000: LET l=l+1: LET o$="co
nquista de": GO SUB 7000: LET l=
l+2: INVERSE 1: LET o$="C E U T
A ": GO SUB 7000: INVERSE 0
2005 PRINT AT 7,12: PAPER 2: INK
6: FLASH 1: CHR$ 157
2007 PAUSE 35: PRINT AT 2,31: PA
PER 2: INK 6: FLASH 1: CHR$ 157
2010 PAUSE 100
2012 PRINT AT 10,1: PAPER 2: INK
7: FLASH 1: " 1419 "
2013 PAUSE 25
2014 LET l=12: LET o$="Inicio do
povoamento": GO SUB 7000: LET l=
l+1: LET o$="do arquipelago da":
GO SUB 7000: LET l=l+2: PAPER
6: INK 0: LET o$="M A D E I R A"
GO SUB 7000
2027 PAUSE 75: PAPER 5: INK 1: P
LOT 235,166: DRAW -33,-38,1/2: P
AUSE 25: CIRCLE 198,122,7
2030 GO SUB 9500
2050 CLS: REM acores
2051 RANDOMIZE USR 60370
2052 PAPER 0: INK 6: BRIGHT 1
2054 GO TO 2072
2070 GO SUB 9500: GO TO 2082
2072 LET o$="A procura de novas
ilhas passou": LET l=13: GO SUB
7000: LET o$="entao a ser mais i
nsistente...": LET l=l+1: GO S
UB 7000
2074 FOR k=2 TO 16: PRINT AT 7,k
: PAPER 1: INK 6: CHR$ 144: AT 7,k
-1: CHR$ 128: PAUSE 3: NEXT k
2076 FOR u=2 TO 12: PRINT AT 10,
u: PAPER 1: INK 6: CHR$ 144: AT 10,
u-1: CHR$ 128: PAUSE 3: NEXT u
2078 PRINT AT 16,0: INVERSE 1: F
LASH 1: "1437"
2079 PAUSE 40: LET l=17: LET o$=
"Diogo Silves descobre o primeir
o": GO SUB 7000: LET l=l+1: LET
o$="grupo de ilhas do arquipelgo
dos": GO SUB 7000
2080 PAUSE 25: LET l=20: LET o$=
"A C O R E S ": GO SUB 7000
2081 FOR h=2 TO 13: PRINT AT 6,h
: PAPER 1: INK 6: CHR$ 144: AT 6,h
-1: CHR$ 128: PAUSE 3: NEXT h: GO
TO 2055
2082 CLS: INK 0: PAPER 5: BRIGH
T 1
2083 RANDOMIZE USR 60680
2084 PRINT AT 1,3: "> Acores"
2085 PLOT 180,168: DRAW -2,-2,PI
: DRAW 2,2,1: PLOT 184,165: PLOT
201,123
2086 PRINT AT 4,3: "> MADEIRA"
2087 PLOT 197,120: DRAW -2,-2,PI

```

```

: DRAW 2,2,1: PLOT 200,124: PLOT
201,123
2091 PAPER 6: INK 0: LET l=9: LE
T o$="A POSSE DESTAS ILHAS FAVOR
ECE A ": GO SUB 7000: PAPER 0: I
NK 6
2092 PAPER 6: INK 0: LET l=l+1:
LET o$="EXPANSAO PORTUGUE
SA": GO SUB 7000
2100 PAPER 0: INK 6
2101 PAUSE 50: PRINT AT 1,1: INK
2: PAPER 6: FLASH 1: "A": AT 2,24
,"A"
2102 LET l=12: LET o$="A-Permite
contornar a zona de ": GO SUB
7000: LET l=l+1: LET o$="": Peri
go dos barcos arabes no ": GO
SUB 7000: LET l=l+1: LET o$="": e
streito de Gibraltar.
GO SUB 7000
2124 PRINT AT 3,26: FLASH 1: CHR$
156: AT 4,27: CHR$ 156: AT 5,26: CH
R$ 156
2126 PAPER 5: INK 0: PAUSE 50: P
LOT 215,159: DRAW -28,6
2128 PRINT AT 6,23: INK 2: PAPER
6: FLASH 1: "M": AT 4,1: "M"
2130 PAPER 0: INK 6: PAUSE 100:
LET l=16: LET o$="M-Oference uma
base de apoio no ": GO SUB 7000
: LET l=l+1: LET o$="alto mar
para as caravelas ": GO SUB 7
000: LET l=l+1: LET o$="que se
dirigem para sul. ": GO SU
B 7000
2131 INK 0: PAPER 5
2132 PAUSE 75: PLOT 183,159: DRA
W 10,-32
2410 GO SUB 9500
2422 GO SUB 9450: RETURN
2501 RANDOMIZE USR 61450
2502 LET l=14: LET o$="A Africa,
ao sul do Cabo Bojador,": GO SUB
7000: LET l=l+1: LET o$="era co
mpletamente desconhecida ": GO
SUB 7000: LET l=l+1: LET o$="par
a os europeus.": GO SUB 7000
2503 PRINT AT 1,1: PAPER 1: BRIG
HT 1: INK 6: CHR$ 164: AT 1,2: "- O
URO
2504 PRINT AT 7,26: INK 6: PAPER
1: BRIGHT 1: CHR$ 164: AT 9,26: CH
R$ 164: AT 10,28: CHR$ 164: AT 10,2
9: CHR$ 164: AT 9,30: CHR$ 164
2505 PAUSE 50: PRINT AT 4,1: PAP
ER 1: BRIGHT 1: INK 7: CHR$ 149: A
T 4,2: "- ESCRAVOS"
2506 PAUSE 50: PRINT AT 8,26: PA
PER 1: INK 5: CHR$ 149: AT 10,27: C
HR$ 149: AT 11,30: CHR$ 149: AT 10,

```

```

31;CHR$ 149: PAUSE 50
2510 PAUSE 75: LET l=18: LET o$="
50" os mercadores arabes conhe-
": GO SUB 7000: LET l=l+1: LET
o$="ciam as rotas do deserto, on
de ": GO SUB 7000: LET l=l+1: L
ET o$="circulavam as suas carava
nas em ": GO SUB 7000: LET l=l+1
: LET o$="busca de ouro e escrav
os.": GO SUB 7000
2540 FOR i=1 TO 6: PRINT AT i,30
: PAPER 1: INK 7;CHR$ 152: PAUSE
25: NEXT i
2543 FOR j=30 TO 25 STEP -1: PRI
NT AT 6,j: PAPER 1: INK 7;CHR$ 1
52: PAUSE 25: NEXT j
2545 PRINT AT 7,29: PAPER 1: INK
7;CHR$ 152:AT 6,28;CHR$ 152:AT
9,27;CHR$ 152
2547 PRINT AT 7,1: PAPER 1: INK
7: BRIGHT 1;CHR$ 152:AT 7,2;"-CA
RAVANA$"
2548 BRIGHT 1
2550 PAUSE 50: LET cox=14: LET a
l=8: GO SUB 8000
2552 PAUSE 50
2555 LET l=14: LET o$="Portugal
estava interessado em ": GO SUB
7000: LET l=l+1: LET o$="encont
rar uma rota maritima para": GO
SUB 7000: LET l=l+1: LET o$="che
gar a esta rica regio.": GO SUB
7000
2560 PAUSE 100: LET l=18: LET o$
="Por isso, depois de terem passa
do": GO SUB 7000: LET l=l+1: LET
o$="o Cabo Bojador (1434), os por
tugues": GO SUB 7000: LET l=l+1:
LET o$="ses continuam as suas vi
agens ao": GO SUB 7000: LET l=l+
1: LET o$="longo da costa de afr
ica.": GO SUB 7000
2562 PAPER 5: INK 0: BRIGHT 1
2565 PAUSE 50: PRINT AT 3,26;CHR
$ 144
2570 PRINT AT 10,1: PAPER 1: INK
7;CHR$ 144:AT 10,2;"-CARAVELAS"
2575 PAUSE 50: FOR j=2550 TO 22
550+10*32 STEP 3*32: FOR i=j TO
j+19: POKE i,109: NEXT i: NEXT j
2580 PAUSE 100
2582 PRINT AT 3,25;CHR$ 144:AT 3
,26: PAPER 6: FLASH 1;"1":AT 0,0
,1
2585 PRINT AT 3,24;CHR$ 144:AT 3
,25;CHR$ 128:AT 0,2: PAPER 2: IN
K 7;"1434":AT 0,7: PAPER 1: INK
7;"CABO BOJADOR"
2590 PAUSE 50: FOR m=4 TO 6: PRI

```

```

NT AT m,24;CHR$ 144:AT m-1,24;CH
R$ 128: PAUSE 50: NEXT m
2600 PAUSE 50: PRINT AT 6,24: PA
PER 6: FLASH 1;"2":AT 3,0;"2": P
RINT AT 3,2: PAPER 2: INK 7;"144
3":AT 3,7: PAPER 1: INK 7;"ARGUI
M"
2610 PAUSE 50: FOR m=7 TO 9: PRI
NT AT m,23;CHR$ 144:AT m-1,23;CH
R$ 128: PAUSE 50: NEXT m: PRINT
AT 9,23;CHR$ 128
2620 PRINT AT 9,24: PAPER 6: FLA
SH 1;"3":AT 5,0;"3": PRINT AT 6,
2: PAPER 2: INK 7;"1444":AT 6,7:
PAPER 1: INK 7;"CABO VERDE"
2630 PAUSE 50: PRINT AT 10,24;CH
R$ 144: PAUSE 50: PRINT AT 11,25
:CHR$ 144:AT 10,24;CHR$ 128
2640 PAUSE 50: PRINT AT 11,26;CH
R$ 144:AT 11,25;CHR$ 128: PAUSE
50: PRINT AT 11,26;CHR$ 128
2650 PRINT AT 11,26: PAPER 6: FL
ASH 1;"4":AT 9,0;"4": PRINT AT 9
,2: PAPER 2: INK 7;"1460":AT 9,7
: PAPER 1: INK 7;"SERRA LEOA"
2660 PAUSE 50: FOR m=28 TO 31: P
RINT AT 12,m;CHR$ 144:AT 12,m-1:
CHR$ 128: PAUSE 50: NEXT m
2670 PAUSE 30: PRINT AT 12,31: P
APER 6: FLASH 1;"5":AT 12,0;"5":
PRINT AT 12,2: PAPER 2: INK 7;"
1470":AT 12,7: PAPER 1: INK 7;"C
OSTA DO OURO"
2700 GO SUB 9500
2990 GO SUB 9450: RETURN
3000 CLS : REM Mina=>Congo
3001 PAPER 0: INK 6: BRIGHT 1: R
ANDOMIZE USR 62070
3002 PAUSE 20: LET l=11: LET o$="
Em 1489, a coroa portuguesa faz
": GO SUB 7000: LET l=l+1: LET o
$="um contrato com um rico merca
dor": GO SUB 7000: PAUSE 50: LET
l=14: LET o$="Segundo o contrat
o, Fernao Gomes": GO SUB 7000: L
ET l=l+1: LET o$="encarregava-se
do descobrimento ": GO SUB 7000
: LET l=l+1: LET o$="de 100 legu
as da costa africana.": GO SUB 7
000: LET l=l+1: LET o$="por ano,
a partir da Serra Leoa.": GO SU
B 7000
3020 PAUSE 50: LET l=19: LET o$="
Como recompensa, Fernao Gomes":
GO SUB 7000: LET l=l+1: LET o$="
ficava com o monopolio comercia
l": GO SUB 7000: LET l=l+1: LET
o$="dessa zona.": GO SUB 7000

```

```

3030 PAPER 5: INK 0
3045 PRINT AT 2,20; PAPER 6; FLA
SH 1;"1";AT 0,0;"1": PRINT AT 0,
2; PAPER 6;" Serra Leoa";AT 0,1
4; PAPER 2; INK 7;"1469"
3050 PAUSE 50: LET COX=10: LET a
l=12: GO SUB 8000
3052 PAUSE 10:
3055 PAPER 0: INK 6: LET l=11: L
ET o$="Pedro Escobar descobre a
famosa": GO SUB 7000: LET l=l+1:
LET o$="zona da MINA, na Costa
do Duro": GO SUB 7000
3057 PAPER 5: INK 0
3050 PAUSE 50: FOR i=22 TO 26: P
RINT AT 2,i;CHR$ 144: PAUSE 80:
PRINT AT 2,i;CHR$ 128: NEXT i
3065 PAUSE 30: PRINT AT 2,26; PA
PER 6; FLASH 1;"2";AT 2,0;"2": P
RINT AT 2,2; PAPER 6;" Mina
";AT 2,14; PAPER 2; INK 7;
"1472"
3067 PAPER 0: INK 6
3070 PAUSE 50: LET l=14: LET o$=
"Construiu-se nessa zona uma im-
": GO SUB 7000: LET l=l+1: LET o
$="portante fortaleza, a que se
deu": GO SUB 7000: LET l=l+1: LE
T o$="o nome de S. Jorge da Mina:
": GO SUB 7000
3075 PAUSE 75: LET l=16: LET o$=
"Essa fortaleza tornou-se a prin-
": GO SUB 7000: LET l=l+1: LET
o$="cipal base para a defesa do
mono": GO SUB 7000: LET l=l+1: L
ET o$="polio portugues de comerc
io do": GO SUB 7000: LET l=l+1:
LET o$="ouro e especiarias afric
anas": GO SUB 7000
3080 PAUSE 25: PRINT AT 1,25; PA
PER 1; INK 7;CHR$ 158;AT 1,26;CH
R$ 150;AT 1,27;CHR$ 158
3090 PAUSE 50: LET COX=10: LET a
l=12: GO SUB 8000
3100 PAUSE 25: LET l=11: LET o$=
"A partir da fortaleza da Mina"
: GO SUB 7000: LET l=l+1: LET o$
="foi possivel organizar viagens
": GO SUB 7000: LET l=l+1: LET o
$="sistematicas de reconhecimento
": GO SUB 7000: LET l=l+1: LET
o$="da costa africana": GO SUB
7000
3105 PAUSE 75: LET l=16: LET o$=
"Numa das expedicoes, em 1482,"
: GO SUB 7000: LET l=l+1: LET o$=
"Diogo Cao atinge a foz do Zaire
": GO SUB 7000: LET l=l+1: LET
o$="o maior rio jamais visto por
um": GO SUB 7000: LET l=l+1: LE

```

```

T o$="europau": GO SUB 7000
3110 PAPER 5: INK 0
3120 PAUSE 50: FOR i=3 TO 4: PRI
NT AT i,26;CHR$ 144: PAUSE 80: P
RINT AT i,26;CHR$ 128: NEXT i
3125 PRINT AT 5,27;CHR$ 144: PAU
SE 80: PRINT AT 5,27;CHR$ 128
3130 PRINT AT 5,28;CHR$ 144: PAU
SE 80: PRINT AT 5,28;CHR$ 128
3135 FOR i=6 TO 8: PRINT AT i,28
:CHR$ 144: PAUSE 80: PRINT AT i,
28;CHR$ 128: NEXT i
3140 PRINT AT 9,29;CHR$ 144: PAU
SE 80: PRINT AT 9,29;CHR$ 128
3145 PRINT AT 9,30;CHR$ 144: PAU
SE 80: PRINT AT 9,30; PAPER 6; F
LASH 1;"3"
3155 PAUSE 20: PRINT AT 4,0; PAP
ER 6; FLASH 1;"3": PRINT AT 4,2;
PAPER 5;" Rio Zaire ";AT 4,14;
PAPER 2; INK 7;"1482"
3200 GO SUB 9500
3210 GO SUB 9450: RETURN
4000 CLS : PAPER 0: BRIGHT 1: IN
K 6
4005 RANDOMIZE USA 62780
4008 PAUSE 15: LET l=11: LET o$=
"Os portugueses penetram no inte
r": GO SUB 7000: LET l=l+1: LET
o$="rior de Africa atraves do ri
o": GO SUB 7000: LET l=l+1: LET
o$="Zaire ": GO SUB 7000
4055 PAUSE 50: LET l=16: LET o$=
"No Zaire, os portugueses estabe
l": GO SUB 7000: LET l=l+1: LET
o$="lecem contactos com um dos m
ais": GO SUB 7000: LET l=l+1: LE
T o$="importantes reinos african
os": GO SUB 7000: LET l=l+1: LE
T o$="- O C O N G O ": GO SU
B 7000
4120 PRINT AT 0,23; PAPER 1; INK
6; BRIGHT 1;CHR$ 150;AT 2,23; I
NK 5;CHR$ 149;AT 1,24;CHR$ 149:
PAUSE 50: PRINT AT 1,21; PAPER 6
: INK 0; FLASH 1;"1"
4125 PRINT AT 0,0; PAPER 1; INK
7;"1": PRINT AT 0,1; PAPER 6; IN
K 0;" ZAIRES ";AT 0,14; PAP
ER 2; INK 7;"1482"
4150 PAUSE 25: LET COX=10: LET a
l=12: GO SUB 8000
4200 PAUSE 15: LET l=11: LET o$=
"Quatro anos mais tarde, em 1486
": GO SUB 7000: LET l=l+1: LET o
$="Diogo Cao avanca mais para su
l": GO SUB 7000: LET l=l+1: LET
o$="ate ao Cabo de Santa Cruz.
": GO SUB 7000
4202 PAPER 5: INK 0

```

```

4205 PAUSE 50: FOR i=3 TO 6: PRI
NT AT i,21;CHR$ 144: PAUSE 80: P
RINT AT i,21;CHR$ 128: NEXT i
4207 PRINT AT 6,22;CHR$ 144: PAU
SE 30: PRINT AT 6,22; PAPER 6; F
LASH 1;"2"
4209 PAPER 0: INK 6
4210 PAUSE 15: LET l=15: LET o$=
"A falta de ventos e de alimento
s": GO SUB 7000: LET l=l+1: LET
o$="obriga-o a desistir da busca
de": GO SUB 7000: LET l=l+1: LE
T o$="uma passagem que permitiss
e che-": GO SUB 7000: LET l=l+1:
LET o$="gar a India ": GO SUB
7000
4220 PRINT AT 3,0; PAPER 1; INK
7;"2": PRINT AT 3,1; PAPER 6; IN
K 0;"NAMISIA ";AT 3,14; PAP
ER 2; INK 7;"1484"
4230 PAUSE 25: LET cox=10: LET a
l=12: GO SUB 8000
4320 PAUSE 10: LET l=11: LET o$=
"No inicio do reinado de D.Joao I
I": GO SUB 7000: LET l=l+1: LET
o$="a tentativa de alcançar a In
dia,": GO SUB 7000: LET l=l+1: L
ET o$="contornando a Africa torn
a-se o": GO SUB 7000: LET l=l+1:
LET o$="principal objectivo do
Estado": GO SUB 7000
4335 PAPER 0: INK 6
4340 LET l=16: LET o$="Pouco dep
ois,Bartolomeu Dias, em": GO SUB
7000: LET l=l+1: LET o$="1488,a
tinge finalmente o extremo": GO
SUB 7000: LET l=l+1: LET o$="sul
do continente africano,": GO SU
B 7000
4345 PAPER 5: INK 0
4350 FOR i=4 TO 7: PRINT AT i,20
;CHR$ 144: PAUSE 30: PRINT AT i,
20;CHR$ 128: NEXT i
4360 PRINT AT 8,21;CHR$ 144: PAU
SE 50: PRINT AT 8,21;CHR$ 128;AT
8,22;CHR$ 144: PAUSE 50: PRINT
AT 8,22;CHR$ 128
4370 FOR i=22 TO 27: PRINT AT 9,
i;CHR$ 144: PAUSE 50: PRINT AT 9,
i;CHR$ 128: NEXT i: PRINT AT 8,
27;CHR$ 144
4380 PAUSE 25: PRINT AT 8,27;CHR
$ 128;AT 7,27; PAPER 6; FLASH 1;
"3"
4400 PAUSE 50: PRINT AT 6,0; PAP
ER 1; INK 7;"3": PRINT AT 6,1; P
APER 6;"C.BOA ESPERANCA";AT 6,1
4; PAPER 2; INK 7;"1488"
4410 GO SUB 9500

```

```

4430 GO SUB 9450: RETURN
5000 CLS: REM India
5010 PAPER 0: INK 6: BRIGHT 1
5012 RANDOMIZE USR 63730
5015 PAUSE 25: LET l=11: LET o$=
"A passagem do Cabo abre a rota"
: GO SUB 7000: LET l=l+1: LET o$=
"para a India das Especiarias."
: GO SUB 7000
5027 PAUSE 25: LET l=14: LET o$=
"as negociacoes do Tratado de":
GO SUB 7000: LET l=l+1: LET o$=
"Torresilhas e a morte de D.Joao II
": GO SUB 7000: LET l=l+1: LET o
$="atrasaram, por mais dez anos,
a": GO SUB 7000: LET l=l+1: LET
o$="etapa final,": GO SUB 7000
5200 PAUSE 25: LET cox=10: LET a
l=12: GO SUB 8000
5205 PAUSE 25: LET l=11: LET o$=
"No final do seculo, em 1498, a"
: GO SUB 7000: LET l=l+1: LET o$=
"armada comandada por Vasco da
": GO SUB 7000: LET l=l+1: LET o$=
"Gama chega a Melinde, na costa
": GO SUB 7000: LET l=l+1: LET o
$="oriental de Africa,": GO SUB
7000
5410 PRINT AT 9,1; PAPER 6; INK
0; FLASH 1;"1";AT 0,19; PAPER 1;
INK 7;"1";AT 0,20; PAPER 6; INK
0; FLASH 0;"Mocambia";AT 0,28;
PAPER 2; INK 7;"1498"
5412 INK 0
5415 PAUSE 50: FOR i=9 TO 7 STEP
-1: PRINT AT i,2; PAPER 5;CHR$
144: PAUSE 40: PRINT AT i,2; PAP
ER 5;CHR$ 128: NEXT i
5420 PRINT AT 6,1; PAPER 6; FLAS
H 1;"2";AT 4,19; PAPER 1; INK 7;
"2";AT 4,20; PAPER 6; INK 0; FLA
SH 0;"Melinde";AT 4,28; PAPER 2
; INK 7;"1498"
5421 PAPER 0: INK 6
5422 LET l=16: LET o$="Um ano de
pois de ter partido de": GO SUB
7000: LET l=l+1: LET o$="Lisboa,
Vasco da Gama chega ao": GO SUB
7000: LET l=l+1: LET o$="porto
de Calecute,na India": GO SUB 70
00: LET l=l+1: LET o$="Atingindo
assim o objectivo que,": GO SUB
7000: LET l=l+1: LET o$="mobili
zara o pais no sec. XV"
5423 GO SUB 7000
5424 PAPER 5: INK 0
5425 PAUSE 50: FOR i=3 TO 3: PRI
NT AT 5,i;CHR$ 144: PAUSE 25: PR
INT AT 5,i;CHR$ 128: NEXT i

```

```

5426 FOR i=9 TO 10: PRINT AT 4,i
CHR$ 144: PAUSE 25: PRINT AT 4,
CHR$ 128: NEXT i
5430 PRINT AT 4,12: PAPER 5: FLA
SH 1: "3": AT 8,19: PAPER 1: INK 7
"3": AT 8,20: PAPER 5: FLASH 0:
INK 0: "Calcute": AT 8,28: PAPER
0: INK 7: "1498"
5510 GO SUB 9500
5512 RESTORE 1000
5520 GO SUB 9450: RETURN
7010 FOR g=1 TO LEN (c$): LET co
=g-1: PRINT AT 1,co;0$(g TO 9):
BEEP .002,2: NEXT g
7020 RETURN
8000 POKE 64983,0: POKE 64986,co
x: POKE 64985,0: POKE 64988,32:
POKE 64989,a1: RANDOMIZE USA 649
82: RETURN

9000 DATA 1,5,0,6,2,4,1,5,3,3,2,
4,4,3,3,3
9010 DATA 5,2,4,3,6,2,5,2,7,2,6,
2,8,2,7,2,9,2,8,2,10,3,9,2,11,4,
10,3,12,5,11,4,12,6,12,5,12,7,12,
6,11,8,12,7,10,9,11,8
9020 DATA 11,9,10,9,12,9,11,9,13
10,12,9,13,11,13,10
9030 DATA 14,10,13,11,15,10,14,1
0,16,10,15,10,17,10,16,10,18,11,
17,10,19,12,18,11,20,12,19,12,21
13,20,12,21,14,21,13
9040 DATA 21,15,21,14,21,16,21,1
5,20,17,21,16,19,16,20,17,18,19,
19,16,17,18,18,19
9050 DATA 18,19,17,18,15,20,16,1
9,14,21,15,20,13,20,14,21,12,20,
13,20
9060 DATA 11,21,12,20,10,22,11,2
1,10,23,10,22,10,24,10,23,10,25,
10,24,10,26,10,25,10,27,10,26,9,
28,10,27,8,29,9,28,8,30,8,29
9200 CLEAR 56499: LOAD "CODE 56
500
9210 RUN
9300 RANDOMIZE USA 64680: RETURN

9450 CLS: BORDER 0: PAPER 0: IN
K 6: RANDOMIZE USA 56500
9455 RETURN
9500 PRINT #1: PAPER 2: FLASH 1:
INK 5: "T - texto" M -
mapa
9503 PAUSE 0
9505 LET e$=INKEY$: IF e$="t" TH
EN GO SUB 9300: RETURN
9507 IF e$="m" THEN GO SUB 9450:
GO TO 1050
9509 IF e$=CHR$ 13 THEN GO TO 95
03

```

```

9510 GO TO 9503
9990 PRINT (PEEK 23653+(PEEK 236
54)*256)-(PEEK 23635+(PEEK 23636
)*256)

```

O código máquina

Se o BASIC do programa sobre a expansão é razoavelmente longo, a parte «máquina» é-o muito mais. Esse facto deve-se à circunstância de o programa ter muitos gráficos animados, mapas, etc., e que constitui, de resto, parte considerável do interesse e qualidade do mesmo.

Inicialmente, considerei a hipótese de não fazer a listagem dessa parte do programa, já que é constituída por um número invulgar de instruções e levará certamente muito boas horas a ser introduzida. Aliás, caso alguém queira introduzir apenas o BASIC, recorrendo depois ao programa «Paint Box» (já referido) para a criação dos mapas e outros gráficos, poderá fazê-lo.

Todavia, pareceu-me vantajoso incluir a listagem referida, tanto mais que é fornecido — no capítulo sobre programação — um programa que permite introduzir o código máquina de forma fácil. Além do que nada obriga a que a sua teclagem seja feita de uma só vez.

A opção é do leitor. Contudo (e estará certamente recordado da qualidade de alguns dos gráficos apresentados no capítulo sobre aplicações), aconselho-o a gastar esse tempo, uma vez que os resultados são extraordinários, nomeadamente no que respeita à animação. Para além do facto de este programa poder ser utilizado muitas vezes e por muitos professores — factor que acaba por tornar o trabalho inicial muito compensatório.

Por razões de ordem técnica (sobreposição do Desassemblador com os endereços dos 9K dos gráficos) a listagem indica como início o endereço 36500, seguido depois até ao 45535. Todavia, ao fazer a introdução os códigos hexadecimais, inicie a colocação no endereço 56500 e seguintes, até ao 65535. (Não há dificuldade, uma vez que o carregador é automático e, de qualquer modo, torna-se bastante fácil verificar erros já que lhe basta somar 20000 ao número indicado para obter o endereço final correcto. Assim, por exemplo, o endereço indicado como 38985 é, na realidade, o 58985.)

Devido à extensão dos códigos, omite-se uma vasta zona de endereços, na impressão do livro: entre o número 43363 (endereço real 63363) e o número 43729 (endereço real 63729), o mesmo sucedendo entre o número 44770 (endereço real 64770) e o número 44808 (endereço real 64808) e ainda entre o número 45256 (endereço real) e o número 45293 (endereço real 65293). Em todos estes espaços — ou seja, nas zonas que vão entre cada dois dos endereços acima indicados (e que não aparecem na listagem) — deve ser introduzido o código hexadecimal 00.

197

37710	01	FF	06	37855	01	FE	01	38000	01	FF	00	38145	01	FF	00	38290	01	FF	00	38435	01	FF	00	38580	01	FF	00	38725	01	FF	00	38870	01	FF	00	39015	01	FF	00	39160	01	FF	00	39305	01	FF	00	39450	01	FF	00	39595	01	FF	00	39740	01	FF	00	39885	01	FF	00	40030	01	FF	00	40175	01	FF	00	40320	01	FF	00	40465	01	FF	00	40610	01	FF	00	40755	01	FF	00	40900	01	FF	00	41045	01	FF	00	41190	01	FF	00	41335	01	FF	00	41480	01	FF	00	41625	01	FF	00	41770	01	FF	00	41915	01	FF	00	42060	01	FF	00	42205	01	FF	00	42350	01	FF	00	42495	01	FF	00	42640	01	FF	00	42785	01	FF	00	42930	01	FF	00	43075	01	FF	00	43220	01	FF	00	43365	01	FF	00	43510	01	FF	00	43655	01	FF	00	43800	01	FF	00	43945	01	FF	00	44090	01	FF	00	44235	01	FF	00	44380	01	FF	00	44525	01	FF	00	44670	01	FF	00	44815	01	FF	00	44960	01	FF	00	45105	01	FF	00	45250	01	FF	00	45395	01	FF	00	45540	01	FF	00	45685	01	FF	00	45830	01	FF	00	45975	01	FF	00	46120	01	FF	00	46265	01	FF	00	46410	01	FF	00	46555	01	FF	00	46700	01	FF	00	46845	01	FF	00	46990	01	FF	00	47135	01	FF	00	47280	01	FF	00	47425	01	FF	00	47570	01	FF	00	47715	01	FF	00	47860	01	FF	00	48005	01	FF	00	48150	01	FF	00	48295	01	FF	00	48440	01	FF	00	48585	01	FF	00	48730	01	FF	00	48875	01	FF	00	49020	01	FF	00	49165	01	FF	00	49310	01	FF	00	49455	01	FF	00	49600	01	FF	00	49745	01	FF	00	49890	01	FF	00	50035	01	FF	00	50180	01	FF	00	50325	01	FF	00	50470	01	FF	00	50615	01	FF	00	50760	01	FF	00	50905	01	FF	00	51050	01	FF	00	51195	01	FF	00	51340	01	FF	00	51485	01	FF	00	51630	01	FF	00	51775	01	FF	00	51920	01	FF	00	52065	01	FF	00	52210	01	FF	00	52355	01	FF	00	52500	01	FF	00	52645	01	FF	00	52790	01	FF	00	52935	01	FF	00	53080	01	FF	00	53225	01	FF	00	53370	01	FF	00	53515	01	FF	00	53660	01	FF	00	53805	01	FF	00	53950	01	FF	00	54095	01	FF	00	54240	01	FF	00	54385	01	FF	00	54530	01	FF	00	54675	01	FF	00	54820	01	FF	00	54965	01	FF	00	55110	01	FF	00	55255	01	FF	00	55400	01	FF	00	55545	01	FF	00	55690	01	FF	00	55835	01	FF	00	55980	01	FF	00	56125	01	FF	00	56270	01	FF	00	56415	01	FF	00	56560	01	FF	00	56705	01	FF	00	56850	01	FF	00	56995	01	FF	00	57140	01	FF	00	57285	01	FF	00	57430	01	FF	00	57575	01	FF	00	57720	01	FF	00	57865	01	FF	00	58010	01	FF	00	58155	01	FF	00	58300	01	FF	00	58445	01	FF	00	58590	01	FF	00	58735	01	FF	00	58880	01	FF	00	59025	01	FF	00	59170	01	FF	00	59315	01	FF	00	59460	01	FF	00	59605	01	FF	00	59750	01	FF	00	59895	01	FF	00	60040	01	FF	00	60185	01	FF	00	60330	01	FF	00	60475	01	FF	00	60620	01	FF	00	60765	01	FF	00	60910	01	FF	00	61055	01	FF	00	61200	01	FF	00	61345	01	FF	00	61490	01	FF	00	61635	01	FF	00	61780	01	FF	00	61925	01	FF	00	62070	01	FF	00	62215	01	FF	00	62360	01	FF	00	62505	01	FF	00	62650	01	FF	00	62795	01	FF	00	62940	01	FF	00	63085	01	FF	00	63230	01	FF	00	63375	01	FF	00	63520	01	FF	00	63665	01	FF	00	63810	01	FF	00	63955	01	FF	00	64100	01	FF	00	64245	01	FF	00	64390	01	FF	00	64535	01	FF	00	64680	01	FF	00	64825	01	FF	00	64970	01	FF	00	65115	01	FF	00	65260	01	FF	00	65405	01	FF	00	65550	01	FF	00	65695	01	FF	00	65840	01	FF	00	65985	01	FF	00	66130	01	FF	00	66275	01	FF	00	66420	01	FF	00	66565	01	FF	00	66710	01	FF	00	66855	01	FF	00	67000	01	FF	00	67145	01	FF	00	67290	01	FF	00	67435	01	FF	00	67580	01	FF	00	67725	01	FF	00	67870	01	FF	00	68015	01	FF	00	68160	01	FF	00	68305	01	FF	00	68450	01	FF	00	68595	01	FF	00	68740	01	FF	00	68885	01	FF	00	69030	01	FF	00	69175	01	FF	00	69320	01	FF	00	69465	01	FF	00	69610	01	FF	00	69755	01	FF	00	69900	01	FF	00	70045	01	FF	00	70190	01	FF	00	70335	01	FF	00	70480	01	FF	00	70625	01	FF	00	70770	01	FF	00	70915	01	FF	00	71060	01	FF	00	71205	01	FF	00	71350	01	FF	00	71495	01	FF	00	71640	01	FF	00	71785	01	FF	00	71930	01	FF	00	72075	01	FF	00	72220	01	FF	00	72365	01	FF	00	72510	01	FF	00	72655	01	FF	00	72800	01	FF	00	72945	01	FF	00	73090	01	FF	00	73235	01	FF	00	73380	01	FF	00	73525	01	FF	00	73670	01	FF	00	73815	01	FF	00	73960	01	FF	00	74105	01	FF	00	74250	01	FF	00	74395	01	FF	00	74540	01	FF	00	74685	01	FF	00	74830	01	FF	00	74975	01	FF	00	75120	01	FF	00	75265	01	FF	00	75410	01	FF	00	75555	01	FF	00	75700	01	FF	00	75845	01	FF	00	75990	01	FF	00	76135	01	FF	00	76280	01	FF	00	76425	01	FF	00	76570	01	FF	00	76715	01	FF	00	76860	01	FF	00	77005	01	FF	00	77150	01	FF	00	77295	01	FF	00	77440	01	FF	00	77585	01	FF	00	77730	01	FF	00	77875	01	FF	00	78020	01	FF	00	78165	01	FF	00	78310	01	FF	00	78455	01	FF	00	78600	01	FF	00	78745	01	FF	00	78890	01	FF	00	79035	01	FF	00	79180	01	FF	00	79325	01	FF	00	79470	01	FF	00	79615	01	FF	00	79760	01	FF	00	79905	01	FF	00	80050	01	FF	00	80195	01	FF	00	80340	01	FF	00	80485	01	FF	00	80630	01	FF	00	80775	01	FF	00	80920	01	FF	00	81065	01	FF	00	81210	01	FF	00	81355	01	FF	00	81500	01	FF	00	81645	01	FF	00	81790	01	FF	00	81935	01	FF	00	82080	01	FF	00	82225	01	FF	00	82370	01	FF	00	82515	01	FF	00	82660	01	FF	00	82805	01	FF	00	82950	01	FF	00	83095	01	FF	00	83240	01	FF	00	83385	01	FF	00	83530	01	FF	00	83675	01	FF	00	83820	01	FF	00	83965	01	FF	00	84110	01	FF	00	84255	01	FF	00	84400	01	FF	00	84545	01	FF	00	84690	01	FF	00	84835	01	FF	00	84980	01	FF	00	85125	01	FF	00	85270	01	FF	00	85415	01	FF	00	85560	01	FF	00	85705	01	FF	00	85850	01	FF	00	85995	01	FF	00	86140	01	FF	00	86285	01	FF	00	86430	01	FF	00	86575	01	FF	00	86720	01	FF	00	86865	01	FF	00	87010	01	FF	00	87155	01	FF	00	87300	01	FF	00	87445	01	FF	00	87590	01	FF	00	87735	01	FF	00	87880	01	FF	00	88025	01	FF	00	88170	01	FF	00	88315	01	FF	00	88460	01	FF	00	88605	01	FF	00	88750	01	FF	00	88895	01	FF	00	89040	01	FF	00	89185	01	FF	00	89330	01	FF	00	89475	01	FF	00	89620	01	FF	00	89765	01	FF	00	89910	01	FF	00	90055	01	FF	00	90200	01	FF	00	90345	01	FF	00	90490	01	FF	00	90635	01	FF	00	90780	01	FF	00	90925	01	FF	00	91070	01	FF	00	91215	01	FF	00	91360	01	FF	00	91505	01	FF	00	91650	01	FF	00	91795	01	FF	00	91940	01	FF	00	92085	01	FF	00	92230	01	FF	00	92375	01	FF	00	92520	01	FF	00	92665	01	FF	00	92810	01	FF	00	92955	01	FF	00	93100	01	FF	00	93245	01	FF	00	93390	01	FF	00	93535	01	FF	00	93680	01	FF	00	93825	01	FF	00	93970	01	FF	00	94115	01	FF	00	94260	01	FF	00	94405	01	FF	00	94550	01	FF	00	94695	01	FF	00	94840	01	FF	00	94985	01	FF	00	95130	01	FF	00	95275	01	FF	00	95420	01	FF	00	95565	01	FF	00	95710	01	FF	00	95855	01	FF	00	96000	01	FF	00	96145	01	FF	00	96290	01	FF	00	96435	01	FF	00	96580	01	FF	00	96725	01	FF	00	96870	01	FF	00	97015	01	FF	00	97160	01	FF	00	97305	01	FF</
-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	----	----	-------	----	------

201

207

209

211


```

10 BORDER 0: PAPER 0: CLS : IN
K 4: BRIGHT 1
12 LET co=0
15 LET sc=0
17 DIM a$(11,13)
18 RESTORE
20 FOR a=1 TO 10: READ a$(a):
NEXT a
30 DATA "D. Joao I","D. Pedro
I","D. Joao II","D. Duarte","D.
Manuel I","D. Sebastiao","D. Joa
o III","D. Afonso V","D. Joao II
I","D. Pedro","D. Manuel I"
80 GO TO 1000
105 CLS: PRINT AT 2,11; INVERS
E 1;"M E N U"
110 PRINT AT 6,4;"1 - Cronologi
a"
120 PRINT AT 10,4;"2 - Mapa"
130 PRINT AT 14,4;"3 - Economia
e Demografia"
135 PAUSE 0
140 LET e%=INKEY$: LET e=CODE e
%-48
145 IF e=1 THEN LET e=225: GO S
UB 9000
150 IF e=2 THEN LET e=198: GO S
UB 9000
155 IF e=3 THEN LET e=171: GO S
UB 9000
160 IF e=0 THEN CLS: RETURN
160 GO TO 100
1000 LET co=co+1
1005 LET no=1: GO SUB 6000: LET
q$="A expansao portuguesa comeco
u no reinado de ...": GO SUB 6100
1010 LET at=5: LET li=10: GO SUB
5000
1012 GO SUB 6300
1015 GO SUB 2000: IF r=0 AND co>
=3 THEN LET e=198: GO SUB 9000:
LET cox=13: LET coy=0: LET la=13
: GO SUB 6500: PAUSE 0: LET e=22
5: GO SUB 9000: LET cox=13: LET
coy=10: LET la=5: GO SUB 6500: P
AUSE 200: CLS: GO TO 1000
1017 IF r=0 THEN CLS: GO SUB 10
0: GO TO 1000
1018 IF r<>1 THEN CLS: GO SUB 3
000: GO SUB 100: GO TO 1000
1020 CLS: PRINT "RESPOSTA CORRE
CTA": LET sc=sc+10: PAUSE 100: C
LS
1030 LET co=0
1050 LET co=co+1: LET no=2: GO S
UB 6000: LET q$="O primeiro rei

```

```

a beneficiar dos contactos com o
Oriente foi ...": GO SUB 6100:
LET at=3: LET li=10: GO SUB 5000
1051 GO SUB 6300
1053 GO SUB 2000: IF r=0 AND co>
=3 THEN LET e=198: GO SUB 9000:
LET cox=13: LET coy=19: LET la=1
3: GO SUB 6500: PAUSE 0: LET e=2
25: GO SUB 9000: LET cox=17: LET
coy=10: LET la=5: GO SUB 6500:
PAUSE 200: CLS: GO TO 1050
1054 GO SUB 2000: IF r=0 THEN CL
S: GO SUB 100: GO TO 1050
1055 IF r<>3 THEN CLS: GO SUB 3
000: GO SUB 100: GO TO 1050
1060 CLS: PRINT "RESPOSTA CORRE
CTA": LET sc=sc+10: PAUSE 100: C
LS
1090 LET co=0
1100 LET co=co+1: LET no=3: GO S
UB 6000: LET q$="Os portugueses
receberam Macau no reinado de .
...": GO SUB 6100: LET at=3: LET
li=10: GO SUB 5000
1102 GO SUB 6300
1103 GO SUB 2000: IF r=0 AND co>
=3 THEN LET e=198: GO SUB 9000:
LET cox=21: LET coy=19: LET la=1
3: GO SUB 6500: PAUSE 0: LET e=2
25: GO SUB 9000: LET cox=18: LET
coy=10: LET la=5: GO SUB 6500:
PAUSE 200: CLS: GO TO 1100
1104 IF r=0 THEN CLS: GO SUB 10
0: GO TO 1100
1105 IF r<>5 THEN CLS: GO SUB 3
000: GO SUB 100: GO TO 1100
1110 CLS: PRINT "RESPOSTA CORRE
CTA": LET sc=sc+10: PAUSE 100: C
LS
1200 REM *** nivel 2 ***
1500 DIM a$(9,25)
1502 CLS: RESTORE 1510
1505 FOR a=1 TO 9: READ a$(a): N
EXT a
1510 DATA "As especiarias do Ori
ente","O Ouro da Mina","O comerc
io de Escravos","O Reino","O Acu
car da Madeira","O Pau-Brasil","
A Alfandega de Lisboa","Os Acore
s","As especiarias do Oriente"
1600 LET co=0
1605 LET no=3: LET co=co+1: GO S
UB 6000: LET q$="Em 1506, a prin
cipal fonte de receitas do Est
ado era...": GO SUB 6100
1610 LET at=5: LET li=1: GO SUB
5000
1615 GO SUB 6300
1620 GO SUB 2000: IF r=0 AND co>

```

```

=3 THEN LET e=171: GO SUB 9000:
LET cox=2: LET coy=16: LET la=11
: GO SUB 6500: PAUSE 0: CLS : GO
TO 1605
1625 IF r=0 THEN CLS : GO SUB 10
0: GO TO 1605
1635 IF r<>4 THEN CLS : GO SUB 3
000: GO SUB 100: GO TO 1605
1640 CLS : PRINT "RESPOSTA CORRE
CTA": LET sc=sc+10: PAUSE 100: C
LS
1650 CLS : PRINT "nivel 3"
1999 STOP
2000 LET r%=INKEY$: LET r=CODE r
%-48: IF r%="" THEN GO TO 2000
2010 RETURN
3000 PRINT AT 10,7: FLASH 1:" E
R R A D O ";AT 20,0:"consulte o
mapa 2 a cronologia ": PAUSE 200
: RETURN
5000 LET g=1: FOR k=6 TO 10: PRI
NT AT k,1:g:" - ";a$(k-at): LE
T g=g+1: NEXT k
5010 RETURN
6000 PRINT PAPER 5: INK 0;"QUEST
AO ";no: RETURN
6100 PRINT
6110 FOR q=1 TO LEN (q%): PRINT
q$(q TO q): BEEP .002,q: NEXT q
: RETURN
6300 LET t$=" Entra o No da res
posta certa ou
Tecla '0' para mai
s informacoes": PRINT AT 19,0: P
APER 1: INK 7:t$: RETURN
6500 REM *** atributos****
6510 LET al=1: LET atr=112: LET
rnd=64982: POKE rnd+1,atr: POKE
rnd+4,cox: POKE rnd+3,coy: POKE
rnd+6,la: POKE rnd+7,al: RANDOMI
ZE USR rnd: RETURN
9000 CLS : POKE 64549,28: POKE 6
4550,e: RANDOMIZE USR 64548: PAU
SE 0
9010 RETURN
9900 CLEAR 43803: LOAD ""CODE 43
804
9990 RUN
9995 LET stk=PEEK 23653+255*(PEE
K 23654): LET rmt=PEEK 23730+255
*(PEEK 23731)
9996 LET mem=rmt-stk: PRINT "mem
oria livre=";mem;"bytes": STOP
9999 SAVE "teste4" LINE 9900: SA
VE "code"CODE 43804,65535-43804

```

Listagem do programa de «ESTATÍSTICA»
apresentado para a disciplina de Economia

```

500 CLS
510 PRINT TAB 3;"E S T A T I S
T I C A S"
520 PRINT AT 6,0
530 PRINT "(1) INTRODUCAO DE NO
VOS DADOS"
540 PRINT "(2) GERAR DADOS ALEA
TORIOS"
550 PRINT "(3) ALTERAR DADOS"
560 PRINT "(4) SAVE/LOAD DADOS"
570 PRINT "(5) CALCULO DE ESTAT
ISTICAS"
580 PRINT "(6) HISTOGRAMA"
590 PRINT "(7) TERMINAR"
600 PRINT AT 21,0:"INDIQUE NUME
RO DA OPCAO"
610 INPUT sel
620 IF sel=1 THEN GO SUB 3000
630 IF sel=2 THEN GO SUB 1000
640 IF sel=3 THEN GO SUB 4000
650 IF sel=4 THEN GO SUB 1500
660 IF sel=5 THEN GO SUB 5500
670 IF sel=6 THEN GO SUB 6000
680 IF sel=7 THEN STOP
690 GO TO 500
1000 CLS
1002 LET cont=0
1010 PRINT "DADOS ALEATORIOS"
1020 PRINT "NUMERO DE VALORES:";
1030 INPUT n
1040 PRINT n
1050 DIM d(n)
1060 PRINT AT 3,0:" DADOS FRACCI
ON./INTEIROS i/i ";
1070 INPUT a$
1080 IF a$<>"f" AND a$<>"i" THEN
GO TO 1050
1090 PRINT a$
1100 LET t=0
1110 IF a$="i" THEN LET t=1
1120 PRINT AT 4,0:" VALOR MINIMO
"
1130 INPUT l
1140 PRINT l
1150 PRINT " VALOR MAXIMO ";
1160 INPUT h
1170 PRINT h
1180 IF h>l THEN GO TO 1210
1190 PRINT "VALOR MAXIMO<VALOR M
INIMO !"
1200 GO TO 1120
1210 FOR i=1 TO n
1212 LET cont=cont+1
1214 IF cont=20 THEN GO SUB 6222

```

```

:CLS
1220 LET d(i)=RND*(h-l+t)+1
1230 IF t=1 THEN LET d(i)=INT d(i)
1240 PRINT "valor do dado ";i;"
      =";d(i)
1250 NEXT i
1260 LET cont=20: GO SUB 6222
1270 GO TO 8900
1500 CLS
1510 PRINT "SAVE ou LOAD dados s
      /l?"
1520 INPUT a$
1530 IF a$<>"l" AND a$<>"s" THEN
      GO TO 1500
1540 IF a$="l" THEN GO TO 1600
1550 PRINT "nome arquivo"
1560 INPUT f$
1570 SAVE f$ DATA d()
1580 GO TO 8900
1600 PRINT "TEM A CERTEZA QUE QU
      ER CARREGAR?"
1610 INPUT a$
1620 IF a$="n" THEN GO TO 8900
1630 IF a$<>"s" THEN GO TO 1600
1640 PRINT "NOME?"
1650 INPUT f$
1660 LOAD f$ DATA d()
1670 LET t=d(1)
1680 LET d(1)=PEEK 23629+256*PEE
      K 23630
1690 LET n=PEEK (d(1)-1)+256*PEE
      K d(1)
1700 LET d(1)=t
1710 RETURN
2000 LET m=0
2010 LET s=0
2020 LET l=d(1)
2030 LET h=l
2040 FOR i=1 TO n
2050 LET m=m+d(i)
2060 IF l>d(i) THEN LET l=d(i)
2070 IF h<d(i) THEN LET h=d(i)
2080 NEXT i
2090 LET m=m/n
2100 FOR i=1 TO n
2110 LET s=s+(d(i)-m)*(d(i)-m)
2120 NEXT i
2130 LET s=s/(n-1)
2140 RETURN
2500 CLS
2510 PRINT "Numero de valores="
      ;n
2520 PRINT "Maximo=";h
2530 PRINT "Minimo=";l
2540 PRINT "Gama=";h-l
2550 PRINT "Media=";m
2560 PRINT "Variância=";s
2570 PRINT "Desvio Standard=";S
      QRT(s)

```

```

2572 LET cont=20: GO SUB 6222
2580 GO TO 8900
3000 CLS
3002 LET cont=0
3010 PRINT "Introducao de dados"
3020 PRINT "Quantos valores?";
3030 INPUT n
3040 CLS: PRINT "N. de valores
      ";n
3042 PRINT
3050 DIM d(n)
3060 FOR i=1 TO n
3062 LET cont=cont+1
3064 IF cont=20 THEN GO SUB 6222
3070 PRINT "Valor ";i;" = ";
3080 INPUT d(i)
3090 PRINT d(i)
3100 NEXT i
3102 LET cont=20: GO SUB 6222
3110 PRINT "INTRODUCAO DE DADOS
      COMPLETA"
3120 GO TO 8900
4000 CLS
4010 PRINT TAB 5;"Corr. dados"
4020 PRINT AT 5,0
4030 PRINT "(1) Listar dados"
4040 PRINT "(2) Alterar dados"
4050 PRINT "(3) Apagar dados"
4060 PRINT "(4) Acrescentar dado
      s"
4070 PRINT "(5) Voltar ao menu p
      rincipal"
4080 PRINT AT 21,0;"INDIQUE NUME
      RO DA OPCAO"
4090 INPUT ed
4100 IF ed=1 THEN GO SUB 4200
4110 IF ed=2 THEN GO SUB 4500
4120 IF ed=3 THEN GO SUB 4600
4130 IF ed=4 THEN GO SUB 4800
4140 IF ed=5 THEN RETURN
4150 GO TO 4000
4200 CLS
4202 LET cont=0
4210 PRINT "Inicio da listagem :
      ";
4220 INPUT l
4230 PRINT l
4240 PRINT "Listagem termina em
      (-1 ate ao fim) ";
4250 INPUT h
4260 PRINT h: PRINT
4270 IF h<0 THEN LET h=n
4280 IF l>h THEN GO TO 4200
4290 IF l>n OR h>n OR l<1 OR h<1
      THEN GO TO 4200
4300 FOR i=l TO h
4302 LET cont=cont+1
4304 IF cont=20 THEN GO SUB 6222
      :CLS
4310 PRINT "Valor ";i;" = ";d(i)

```


Os blocos do programa estão assim constituídos:

<i>N.º da linha</i>	<i>descrição</i>
500 — 690	Menu principal
1000 — 1270	gerador de dados
1500 — 1710	SAVE e LOAD de dados
2000 — 2140	cálculo estatístico
2500 — 2580	impressão de resultados
3000 — 3120	introdução de dados
4000 — 4150	alteração de menu
4200 — 4330	listagem de dados
4500 — 4590	alteração de dados
4600 — 4790	apagamento de dados
4800 — 5020	adição de dados
5500 — 6230	desenho de histogramas
7000 — 7110	construção de frequências

ANEXO B

Obras úteis

Sem qualquer atitude parcial, tem de reconhecer-se que a Editorial Presença foi não só a pioneira no lançamento em quantidade e qualidade de obras dedicadas à microinformática como a que apoiou — antes de qualquer outra — o lançamento de autores nacionais, produzindo livros sobre o assunto.

É, pois, natural, que seja entre as suas edições que pode encontrar-se a maioria das obras úteis aos diversos sectores desta actividade. Como é igualmente natural que autores e leitores apoiem esse esforço, garantindo, estou certo, a continuação do mesmo.

Passarei pois a indicar as obras que, dessa editora, poderão ser adquiridas, assinalando com um asterisco as que me parecem mais importantes ou até indispensáveis:

- n. 56 — ABC dos computadores
- n. 130 — Introdução à programação de Microcomputadores
- * — n. 133 — Guia prático de BASIC
- n. 137 — Guia de COBOL
- n. 141 — ABC da programação de computadores
- n. 144 — Como programar o seu ZX 81
- n. 147 — BASIC para microcomputadores
- * — n. 148 — Como programar o seu ZX Spectrum
- n. 153 — Manual do ZX Spectrum
- * — n. 154 — Z80 assembler para o ZX Spectrum
- n. 157 — 26 programas BASIC para microcomputadores
- n. 158 — Aprenda PASCAL no seu microcomputador
- * — n. 161 — Código máquina para principiantes
- n. 162 — Código máquina para programadores avançados
- n. 165 — Guia de FORTRAN
- n. 176 — Guia de PASCAL
- * — n. 177 — O ZX Spectrum na Educação
- * — n. 186 — Processamento de dados

Das obras inglesas não traduzidas, parece-me importante adquirir:

- «The ZX Spectrum and how to get the most from it»
- «An expert guide to the ZX Spectrum» (muito útil)
- «Databases for fun and profit»
- «Make the most of your ZX Microdrive»

No campo das revistas, há vantagem em assinar (ou adquirir semanal ou mensalmente, conforme os casos) as seguintes:

- «Sinclair User» (sempre com as últimas novidades para os computadores da Sinclair)
- «ZX Computing»
- A revista espanhola «Microhobby» (acessível e muito útil)
- A revista Portuguesa «Software»

ANEXO C

GLOSSÁRIO

Acumulador — parte da unidade central de processamento de um computador, onde são armazenados dados intermédios do processamento.

ALGOL (ALGOritmic Language) — linguagem de programação científica, muito próxima da notação matemática usual.

Alfanumérico — Conjunto constituído por letras e números.

Algoritmo — Conjunto de acções elementares (rotinas, no caso da informática) destinadas, globalmente, a solucionar um problema.

ALPHARD — Linguagem utilizada na escrita de sistemas.

Análise (ver «Programação») — Passo que antecede a programação.

APL (A Programming Language) — Linguagem utilizada em programação.

Arquitectura — embora com várias aplicações, utiliza-se, sobretudo, para designar a projecção de sistemas de computador.

Árvore — sistema básico de funcionamento dos computadores, no que respeita à sua parte lógica. A estrutura em árvore tem especial aplicação no sistema de Floppy Disks.

ASCII (American Standard Code for Information Inter-change) — código universal para letras, números e símbolos gráficos, a cada um dos quais é atribuído um número entre 0 e 255.

Assembler — programa que converte um programa escrito em linguagem assembler (intermédia entre o BASIC e a linguagem máquina) num outro de linguagem máquina, directamente colocado na memória do computador.

Array — modo de estruturação de dados destinado a reunir sob um índice o conjunto. Pode ter uma ou mais dimensões.

Banco de dados — conjunto mais ou menos completo de informações sobre uma determinada área (científica ou outra), cobrindo um domínio particular do conhecimento, e directamente exploráveis.

Base de dados — colecção de diversos tipos de informação (dados), tais como nomes, endereços, etc., organizados por forma a poderem ser acedidos e combinados pelo computador.

BASIC (*Beginner's All Purpose Symbolic Instruction Code*) — A mais universal das linguagens de programação. As palavras-chave que a constituem são, genericamente, resultantes do vocabulário inglês correspondente, o que a torna muito fácil de aprender e utilizar. É, todavia, lenta para determinado tipo de operações (gráficos, por exemplo).

Baud — unidade de transferência de dados. Indica o número de bits transferidos por segundo.

Binário — sistema de numeração de base 2, que utiliza apenas os símbolos 0 e 1. Os computadores funcionam com este sistema.

Bit (*binary digit*) — unidade mínima binária da informação (0 ou 1).

BLISS (*Basic Language for Implementation of System Software*) — Linguagem de programação destinada à escrita de sistemas de Software.

Bloco — conjunto de instruções compreendido entre dois limitadores (ver «rotina»).

Buffer — espécie de reservatório entre dois elementos do sistema (computador e impressora, por exemplo), que permite a continuação do trabalho num deles, enquanto se «esvazia» da informação para o outro.

Bug — Erro num programa.

Bus — um fio ou conjunto de fios destinados a ligar o computador a um periférico.

Byte — conjunto de *bits* (oito ou dezasseis), que constitui um carácter.

Cabeça de leitura/escrita — dispositivo através do qual é possível registar e ler informação contida num suporte magnético.

Cabeça de programa — parte indicializadora do programa, que contém indicações sobre o seu comprimento, endereçamento inicial, etc.

Cadeia — Conjunto de dados do mesmo tipo, colocados contigualmente.

Campo — parte de um registo (em ficheiro, por exemplo), previamente definida.

Canal — caminho por onde circula informação, no interior ou para o exterior de um computador.

Caracter — Símbolo reconhecido pelo utilizador, para cuja definição são necessários 8 bits.

Chave — informação alfanumérica que permite identificar um qualquer conjunto de dados.

Ciclo — Sucessão de operações realizadas durante um determinado período e conduzindo a um resultado parcelar previsto.

Circuito impresso — placa rígida em material isolante, em que se encontram canais metálicos que ligam os vários componentes soldados no mesmo.

Circuito integrado (chip) — Circuito electrónico (equivalente a uma parte do circuito impresso) de dimensões reduzidas, destinado a produzir certas operações.

COBOL (*Common Business Oriented Language*) — Linguagem de programação especialmente destinada à gestão de empresas.

Código — conjunto de regras que permitem a representação de dados.

Código de caracteres — código binário que representa um carácter.

Código máquina — código do microprocessador de um computador, utilizando os dígitos 0 e 1 nas suas várias combinações possíveis.

Compilação — transformação de um programa, escrito numa linguagem de alto nível (BASIC, por exemplo), em idêntico programa em linguagem máquina.

Compactação — compressão de um programa.

Comparação — verificação de identidade entre dois dados (a instrução IF dos computadores é especialmente utilizada para o efeito).

Compatibilidade — possibilidade de utilização de um programa específico de um computador em outro (BASIC, ou outra linguagem, idênticas).

Computador — dispositivo electrónico destinado à realização de cálculos, aceitando dados e processando-os.

Concatenar — somar cadeias (*strings*).

Conjunto de caracteres (set) — totalidade dos caracteres possíveis de ser utilizados simultaneamente através do teclado do computador.

Controlador — Periférico de controlo que assegura a gestão dos acessos diferenciados, em determinado momento, a um sistema informático.

Conversão — operação de transformação de dados.

CP/M (Control Program Monitor) — sistema de comando universal que permite, na prática, uma conversão simples de programas.

Cursor — Marca luminosa no écran do TV ou monitor, que indica o local em que pode ser introduzida informação directamente a partir do teclado.

Dado (data) — termo genérico empregue para designar a informação de um computador.

Desassemblar — traduzir um programa (ou localizações de endereços) em linguagem máquina para um programa em linguagem assembly.

Diagrama — esquema lógico dos circuitos de uma máquina ou dos blocos de um programa.

Dígito — carácter representando o valor de um número reduzido a unidades (0,1,2,3,4,5,6,7,8,9).

Disco — suporte de armazenamento magnético, com dimensões variáveis.

Display — operação que consiste em fazer aparecer informação num écran.

Documentação — conjunto de instruções que acompanham um programa.

E.A.C. — ensino assistido por computador.

Écran — unidade de visualização de um terminal.

Editor — programa que permite efectuar a análise de um ficheiro.

Emulador — dispositivo que permite simular um software.

Endereço — número que indica a localização de uma célula de memória de um computador.

Entrada (input) — dado ou programa fornecido a um computador (ver saída).

EPROM (Erasable Programmable Read Only Memory) — dispositivo (chip) que contém informação permanente, cujo acesso é apenas de leitura.

Fiabilidade — probabilidade de funcionamento sem falhas de um

dispositivo de software ou hardware, em condições definidas e durante um período de tempo determinado, medida percentualmente.

Ficheiro (file) — conjunto de registos estruturados, conservado em memória secundária de um sistema informático.

Flag — indicador de prioridades de uma operação.

Fluxograma — esquema escrito da estrutura e execução de um programa.

Formatação — preparação de um suporte físico (wafer de microdrive ou disco de Floppy) com vista a permitir-lhe receber informação.

FORTTRAN (FORmula TRANslator) — linguagem de programação de alto nível, especialmente empregue no âmbito científico.

Fusão (merge) — mistura de dois programas ou partes dos mesmos.

Gramática — conjunto de regras definidoras da sintaxe de uma determinada linguagem.

Hard Copy — informação de computador, impressa em papel.

Hardware — Conjunto de elementos físicos utilizados no processamento da informação.

Hexadecimal — sistema de numeração de base 16, especialmente utilizado na programação em código máquina.

Impressora — periférico que permite a impressão de informação em papel.

Índice — conjunto de dados que descrevem uma estrutura.

Input — (ver entrada).

Instrução — operação elementar de uma linguagem de programação.

Interface — conjunto de regras de adaptação destinadas a permitir que dois sistemas possam trocar informação.

I/O — (input/output) — símbolos de entrada/saída.

Linguagem de alto-nível — linguagem de programação mais próxima dos códigos linguísticos humanos.

Linguagem de baixo nível — linguagem de programação mais próxima da linguagem máquina do computador.

Listagem — texto completo de um programa ou rotina.

LOGO — linguagem de programação muito simples e especialmente desenvolvida para o ensino da matemática.

Megabyte — um milhão de bytes.

Memória — dispositivo que permite armazenar informação destinada a ser utilizada por um computador.

Memória de acesso aleatório (RAM) — parte da memória de um computador que pode ser escrita ou lida pelo programador.

Memória de leitura (ROM) — parte da memória de um computador que apenas pode ser acedida para leitura.

Memorização — operação de transferência de dados para a memória de um computador.

Menu — lista de comandos ou opções disponíveis num determinado passo de um programa.

Microcomputador — computador construído a partir de um único microprocessador (ZX Spectrum, TC 2068 e outros).

Microprocessador — circuito integrado onde se efectuam as operações centrais de um computador.

Minicomputador — maior que o microcomputador, possui uma unidade central que processa 16 ou 32 bits.

Modem — acoplador acústico destinado ao envio e recepção de informação entre dois computadores, via rede telefónica.

Modulação — técnica que consiste em modificar as características de um sinal nos parâmetros definidos por um outro sinal.

Módulo — parte de um programa destinado a executar um conjunto definido de funções.

Monitor de vídeo — terminal de visualização semelhante a um televisor, que possui, contudo, uma definição perfeita da imagem enviada pelo computador.

Network — rede de computadores ligados interactivamente.

Organigrama — representação gráfica de um algoritmo.

Output — (ver saída).

Package — conjunto de rotinas ou programas, constituindo um programa completo.

PASCAL — linguagem de programação de alto nível destinada à programação estruturada.

PILOT — linguagem de alto nível destinada à educação.

Periférico — dispositivo exterior a uma unidade de processamento.

Pixel — a mais pequena superfície homogénea constituinte de uma imagem impressa em screen.

Porto (port) — canal de entrada/saída de informação do e para o computador.

Porta (gate) — circuito capaz de alternar modos de funcionamento na entrada/saída de dados.

Processamento de dados — trabalho executado pelo computador na transformação de informação (data).

Programa — conjunto de instruções destinadas a executar determinada tarefa previamente determinada.

RAM — (ver memória de leitura).

ROM — (ver memória de acesso aleatório).

Rede — (ver network).

Registo — conjunto de informações manipuladas em bloco.

RS-232 — interface standard que permite ligar muitos dos periféricos existentes a um computador.

Robot — dispositivo destinado a executar tarefas físicas, controlado por computador ou processador.

Rotina — sequência de instruções no interior de um programa, as quais executam uma determinada tarefa.

Saída (output) — informação resultante de um processamento realizado por um computador.

Salto (jump) — passagem de um ponto do programa para outro.

Save — guardar informação em periférico magnético.

Simbolo — representação convencional de uma noção.

Sistema operativo — programa ou conjunto de programas que controlam o funcionamento básico do computador.

Software — programas destinados a processamento num computador.

Suporte — qualquer material destinado a guardar informação.

Subrotina — (ver rotina).

Teclado — conjunto completo de teclas que permite entrar em comunicação com o computador.

Terminal — órgão de acesso a um computador.

Tradutor de linguagem — (ver compilador).

ULA (Unidade Aritmética e Lógica) — parte do CPU de um computador onde se realizam operações lógicas e cálculos.

Unidade Central de Processamento (CPU) — unidade de computador onde se realiza o processamento de dados.

Variável — posição de memória representada por um identificador, cujo conteúdo varia no decorrer da execução de um programa.

ANEXO D

Índice de programas

ÍNDICE DOS PROGRAMAS LISTADOS NA OBRA

Análise de frases	102
Ângulos	121
Apagar palavras no écran	153
Áreas	121
Avaliação	132
Base de dados	140
Cálculo e animação dos corpos celestes	168
Carregador de código máquina	65
Catalogador	154
Climas	108
Constelações	105
Conversão de temperaturas	52
Copiador	149
Definição de caracteres	137, 145
Desenhos de precisão	99
Escrita diagonal	138
Escrita de música	114
Espirais	86
Estatística	119, 214
Expansão portuguesa no Séc. XV	74, 185
Ficheiro de alunos	158
Figuras geométricas	56
Gráficos de alta resolução	49
Inputs	154
Inversão do écran	154
Leitura de cabeças	136
Matrizes	96
Mecânica celeste	110
Médias	119, 134

cálculo de expressões	52
divisibilidade	53
equações	58
funções	48, 57, 124
gerador de números pares	57
máximo divisor comum	55
múltiplos	54
números primos	54
polinômios	52
quadrados de números	50, 54
raiz quadrada	54
simulações	48
soma de números	47, 51
Ordenamento alfabético	147
Padrões	128
Piano	116
Polígonos	91
Renumeração	135
Scroll lateral	153
Tamanhos de caracteres	138
Teste à RAM	139
Teste de História	212
Traçado de mapas	103
Vida	112
Vocabulário de Francês	101

PREÂMBULO 9

I PARTE — MONTAGEM DE UM SISTEMA INFORMÁTICO

Cap. 1 — A ESCOLHA DO SISTEMA	13
1.1 — O que é um sistema informático	13
1.2 — Os elementos do sistema	16
1.3 — O sistema eventualmente existente	19
1.4 — O sistema possível/aconselhável	22
Cap. 2 — COMO MONTAR UMA SECÇÃO DE INFOR- MÁTICA	25
2.1 — A sala	25
2.2 — O equipamento	26
2.3 — A instalação	27
2.4 — Alguns cuidados básicos	30
2.5 — A falta de verba... ..	32
Cap. 3 — UM PROJECTO INTERESCOLAS	34
3.1 — Uma rede entre escolas	34
3.2 — Um banco de dados comum	35
3.3 — O futuro	36
RTE — A UTILIZAÇÃO DO SISTEMA	
Cap. 1 — QUE FAZER COM O SISTEMA	39
1.1 — Um novo método de trabalho	39
1.2 — Tempos e ocupações da sala	41
1.3 — Como começar	42

Cap. 2 — ALGUMAS OCUPAÇÕES POSSÍVEIS	43
2.1 — Os cursos básicos	43
2.2 — A programação	58
2.3 — Aulas	66
2.4 — Tempos livres e clube de informática	68
Cap. 3 — APLICAÇÕES PRÁTICAS	73
3.1 — A utilização em algumas disciplinas	73
História	74
Educação visual	85
Línguas	101
Geografia	103
Ciências Naturais	112
Educação Musical	114
Economia	118
Geometria	120
Matemática	124
Trabalhos oficinais	128
Avaliação	132
3.2 — Algumas rotinas e programas úteis	135
3.3 — Programas comerciais	168
Cap. 4 — PROBLEMAS COM O SISTEMA	172
4.1 — As ligações eléctricas	172
4.2 — O televisor e a sintonia	173
4.3 — Guardar e carregar dados	174
4.4 — Periféricos	176
4.5 — Avarias diversas	177
Cap. 5 — ALÉM DO SISTEMA	178
III PARTE — ANEXOS	
Anexo A — Listagens de programas	185
Anexo B — Obras úteis	221
Anexo C — Glossário	223
Anexo D — Índice de programas	231

Fotocomposição de GrafOrigem
Impresso e Acabado na
Tipografia Guerra — Viseu
em 1985
para a Editorial Presença, Lda. --- Lisboa