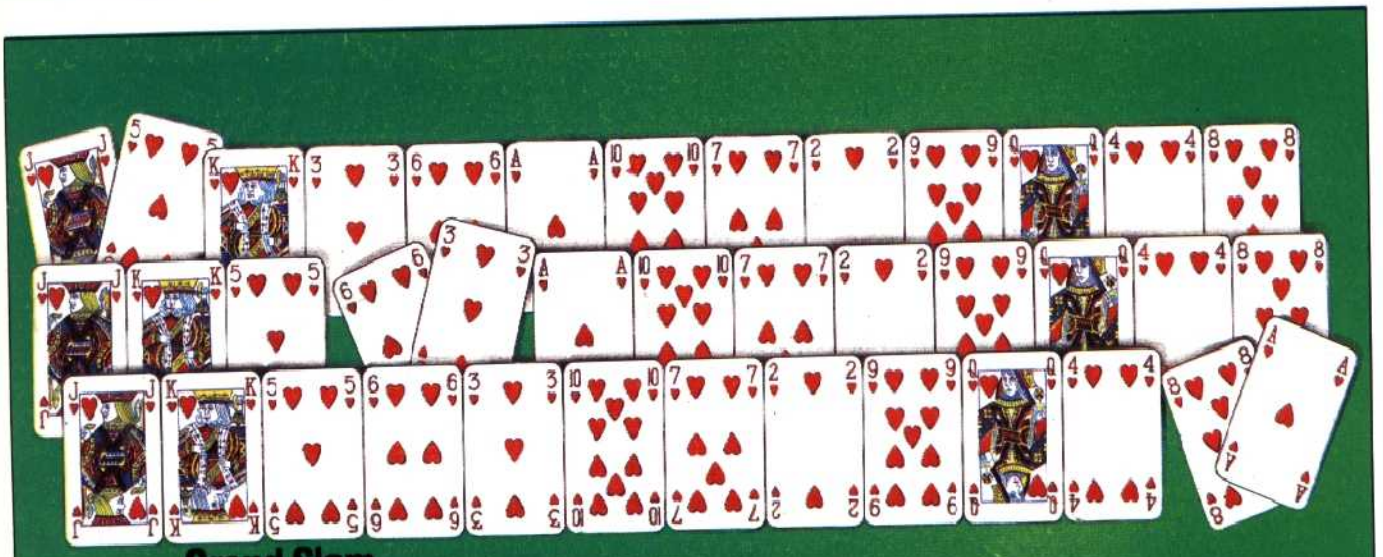

Grand Slam

One way to illustrate a Bubble Sort is with a complete suit of cards that have to be sorted so that the King ends up on the left and the Ace on the right. First the leftmost two cards are compared, and because they are found to be out of order, they are swopped over. Then the second and third cards are compared, and again swopped. By the fifth comparison, this sort method has picked up the Ace, and in all subsequent comparisons, the Ace is swopped from left to right, until at the end of the first 'pass' it has 'bubbled' its way to the right-hand end. By repeating this whole process for the second pass, the two will end up next to the Ace. However, it may take up to 12 such passes before all the cards are in order

TONY LODGE

place the penny card in the gap and proceed to Step 4).

Compare this twopenny card with the penny card (the displaced one). Now repeat Step 3 until the correct position for the penny card is found.

4) Move the penny one position to the right and repeat Steps 2 and 3. When you can't move the penny any further right, the cards will all be in order.

This is called an 'Insertion Sort', and is very similar to the way people sort a hand of cards. Although it is a little harder to program than a Bubble Sort it is a far more efficient method. Later in the course, we will look at some more complex algorithms for sorting data.

```
9  REM**********************
10 REM* SORT ALGORITHMS    *
11 REM**********************
100 INPUT"HOW MANY ITEMS TO BE SORTED";LT
150 IF LT<3 THEN LET LT=3
200 LET LT=INT(LT)
250 DIM R(LT),C(LT)
300 LET Z=0:LET Q=0:LET P=0
350 LET I=1:LET O=0:LET II=2:LET TH=2
400 INPUT"HOW MANY TESTS ";N
450 FOR CT=I TO N
500 GOSUB 4000
550 FOR SR=I TO TH
600 GOSUB 5000
650 PRINT:PRINT:PRINT:PRINT
700 PRINT "TEST #";CT+SR/10
750 INPUT"HIT RETURN TO BEGIN SORT";A$
800 PRINT "THE UNSORTED LIST IS"
850 GOSUB 3000
900 ON SR GOSUB 6000,7000
950 PRINT "THE SORTED LIST IS"
1000 GOSUB 3000
1050 NEXT SR
1100 NEXT CT
1150 END
2999 REM**********************
3000 REM*  PRINT THE LIST    *
3001 REM**********************
3100 FOR K=I TO LT
3200 PRINT R(K);
3300 NEXT K
3400 PRINT
3500 RETURN
3999 REM**********************
4000 REM*   RND GENERATOR    *
4001 REM**********************
4100 RANDOMIZE
4200 FOR K=I TO LT
4300 LET C(K)=INT(100*RND)
4400 NEXT K
4500 RETURN
4999 REM**********************
5000 REM*  RND REGENERATOR   *
5001 REM**********************
5100 FOR K=I TO LT
5200 LET R(K)=C(K)
5300 NEXT K
5400 PRINT:PRINT
5500 RETURN
5999 REM**********************
6000 REM*      BUBBLE        *
6001 REM**********************
6050 PRINT "BUBBLE SORT - GO !!!!!
6100 FOR P=LT-I TO I STEP-I
6150 LET F=-I
6200 FOR Q=I TO P
6250 LET Z=Q+I
6300 IF R(Q)<R(Z) THEN LET D=R(Q):
     LET R(Q)=R(Z):LET R(Z)=D:LET F=0
6350 NEXT Q
6400 IF F=-I THEN LET P=I
6450 NEXT P
6500 PRINT "BUBBLE SORT - STOP !!!!!
6550 RETURN
6999 REM**********************
7000 REM*     INSERTION      *
7001 REM**********************
7050 PRINT "INSERTION SORT - GO !!!!!
7100 FOR P=II TO LT
7200 LET D=R(P)
7300 FOR Q=P TO II STEP-I
7400 LET R(Q)=R(Q-I)
7500 IF D<=R(Q) THEN LET R(Q)=D:LET Q=II
7600 NEXT Q
7700 IF D>R(I) THEN LET R(I)=D
7800 NEXT P
7850 PRINT "INSERTION SORT - STOP !!!!!
7900 RETURN
```

High-Speed Sort

This BASIC program demonstrates the difference in efficiency between a Bubble Sort and Insertion Sort. The code has been written with speed in mind, so we have not documented the operation of the routines. The listing should run on most machines, but see page 215 for ON . . . GOSUB flavours, and page 175 for RND and RANDOMIZE