

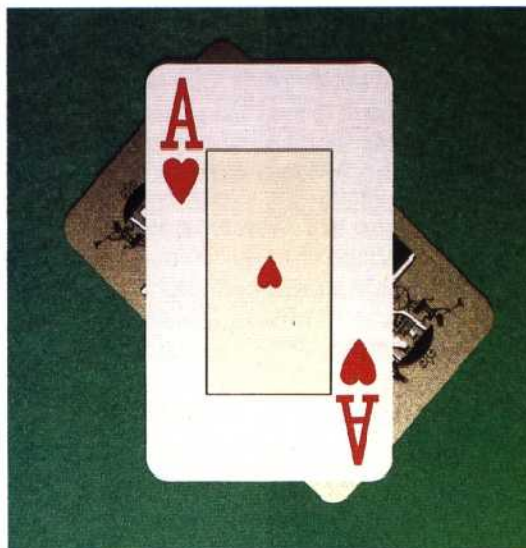
all; but although it is not programmable, it does execute a program. The machine was developed by Plessey in 1973 to replace the original machine, built in 1957. Its function is to randomly generate some 200,000 numbers and write them onto magnetic tape. These numbers are generated from a series starting at the lowest premium bond serial number ever issued, and ending with the highest. The tape is then loaded into an ICL mainframe computer, which compares the numbers with a tape listing the numbers of those bonds that have already been repaid. Once these ineligible numbers have been eliminated, the computer can print out prize warrants and letters to the winners.

Since commissioning, the two ERNIEs have generated the numbers for 22.2 million prizes worth £1,181,843,400. Sounds good? Not really; the chance of a bond winning a prize in the monthly draw is just one in 15,000.

Another often encountered game of chance appears daily in our newspapers. The chances of winning one of the million pound prizes offered in newspaper promotions is even more remote. The numbers on the cards distributed to readers are 12 digits long. A 12 digit sequence running from 000000000000 to 999999999999 offers a million million separate combinations. Statistically, the odds against a particular number coming up on a given morning would be slightly better than a million million to one, since the newspaper concerned publishes two numbers each day. On this basis, it must be extremely doubtful whether

the newspaper will ever have to pay out the big prize.

The situation can best be visualised by imagining a bag containing two and a half million white balls, representing the competitors (the number of cards in circulation), and a million million black balls for the total number of possible combinations. Needless to say, the chances of pulling a white one out the first time are pretty remote. Moreover, the odds do not significantly improve even when a year's worth have been pulled out. Statisticians compute the chances of the newspaper ever having to part with a million pounds at just one in 667.



Twist Of Fate

Pontoon (blackjack or 'vingt-et-un') is available as a game on most home computers, and provides some of the best scope for writing winning programs. The ability to memorise the cards already played increases a player's chances of success, though as casinos won't allow computers at the tables, the celebrated feats have all involved concealed computers (in one case strapped to a player's leg underneath his trousers) or radio links to external machines

Loading The Dice

Gambling's most essential ingredient, random number generation, can be easily simulated on a personal computer. Most versions of BASIC provide a random number generator function. In many cases the numbers so generated are not truly random, however, as the following short program demonstrates:

```
10 LET A = RND
20 LET B = RND
30 LET C = RND
40 PRINT A, B, C
```

In each of the first three lines a supposedly random number is assigned to the variables A, B and C. These are then printed out. This might give the following results (expect yours to differ though):

```
.014007 .964370 .457397
```

But if you re-run the program, most microcomputers will display the same sequence again. What is happening is this: when we ask for RND, the computer responds with the next in a fixed sequence of numbers. Typically this might comprise the one million six-digit fractions between 0.000000 and 0.999999, each occurring once in the complete cycle — but not, of course, in sequence.

Certain BASICS use a slightly different syntax, requiring an expression in parenthesis, called an 'argument'. This takes the form LET A = RND (X). The effect is very similar: RND and RND (X) can both be used in the same way as other variables.

Some BASICS also feature a RANDOMIZE function, which causes the sequence to start at an unpredictable point. Inserting the RANDOMIZE command early on in any program where RND is to be used ensures that a different sequence of numbers will be generated each time the program is RUN.

To simulate the casting of a dice we require integers in the range 1 to 6. It is, however, necessary to eliminate fractions. This is done by using the INTeger function. PRINT INT(6.99) produces the result 6 just as surely as PRINT INT(6.01) does. Anything after the decimal point is entirely discarded.

Since the largest number that RND can generate is .999999 (which, when expressed as an integer, acquires a value of 0), a little multiplication is required. The time honoured formula is:

```
LET A = INT(6 * RND) + 1
```

We multiply by six because a die has six faces. The 'plus one' is simply to ensure that the results range from 1 to 6, and not from 0 to 5.