DANIEL F. MERRIAM, Editor

SPITSYM, A FORTRAN IV COMPUTER PROGRAM FOR SPIT SIMULATION

By

M. J. McCULLAGH

and

C. A. M. KING

University of Nottingham

COMPUTER CONTRIBUTION 50

State Geological Survey

The University of Kansas, Lawrence

1970

in cooperation with the
American Association of Petroleum Geologists
Tulsa, Oklahoma

Editor's Remarks

Simulation is important in developing new concepts in geology. Never before has it been possible to experiment with geological processes and view the results. Much important work has been accomplished in the past several years by J.W. Harbaugh and his coworkers at Stanford University and W.C. Krumbein and his group at Northwestern University.

The first COMPUTER CONTRIBUTION was concerned with simulation of marine sedimentation and was authored by J.W. Harbaugh in 1966. That program later was revised and published as Computer Contribution 9. These publications were followed by W.C. Krumbein's contribution on Markov chains as Computer Contribution 13.

In 1967 a "Colloquium on Simulation" was held on campus at The University of Kansas and the proceedings published as Computer Contribution 22. Subjects included simulation of transgressive and regressive deposits, sedimentation rates, salt domes, a neutron activation system, and reservoir behavior.

The colloquium was followed by two publications, Computer Contributions 24 on a mathematical model for computer simulation of deltaic sedimentation, and Computer Contribution 26 on simulation of transgression and regression models. Computer Contribution 49 is concerned with sedimentary basin simulation.

This publication, Computer Contribution 50, "SPITSYM, a FORTRAN IV computer program for spit simulation", by M.J. McCullagh and C.A.M. King is the latest simulation program. It will be of value to those interested in understanding the geological processes that develop spits. Other similar features undoubtedly can be simulated, and sometime in the future individual features such as delta and spits can be integrated into a program for a realistic development of marine basins.

The program SPITSYM as described here will be made available for a limited time by the Geological Survey for $10.00 (US). An extra charge of $10.00 (US) is made if punched cards are required. The extra charge is to cover postage and handling. A complete list of COMPUTER CONTRIBUTION publications can be obtained by writing Editor, COMPUTER CONTRIBUTIONS, Kansas Geological Survey, The University of Kansas, Lawrence, Kansas 66044.

SPITSYM, A FORTRAN IV COMPUTER PROGRAM FOR SPIT SIMULATION

by

M.J. McCullagh and C.A.M. King

ABSTRACT

The development and operation of a program to simulate the formation of a shingle spit is described. This includes a detailed account of action of the subroutines that simulate the four major processes that are thought to act on the spit. The simulation relies on stochastic techniques and the program is concerned with building a spit in a 50-row by 60-column matrix so that the results of the process are graphically displayed. The example illustrating use of the program is the Hurst Castle spit on the Hampshire coast of England.

HURST CASTLE SPIT

Hurst Castle spit lies on the south coast of England, developed partly in the shelter of the Isle of Wight. The spit is about 2 km long (Fig. 1). It extends straight in a southeasterly direction before gradually turning east-west. One main feature of the spit is a series of recurves that join the main ridge at an acute angle. The recurves face northeast, normal to the elongation of the Solent, an open stretch of water separating the Isle of Wight from the Hampshire mainland. The number and length of the recurves are greater towards the distal end of the spit. Shingle [1] is the predominant material forming the spit, and this is important in considering the processes that are meaningful in its growth.

Formation of the spit has been studied by Lewis (1931, 1938). He showed that the spit was dependent upon the action of several distinct wave types and that other factors entered into its formation. For these reasons this particular spit was thought to be suitable for developing a computer simulation model. The computer program simulates the dominant wave processes and takes into account the other factors important in determining the form of the spit.

Because the spit consists predominantly of shingle, the most permanent features are formed by the action of storm waves. The storm waves, although they comb much shingle offshore from the beach, throw material above the reach of normal waves to form the major ridge that determines the form of the spit. The action of storm waves is one process that is simulated in the program. The available material that the storm waves build into more permanent ridges is derived from the west. The longest fetch to which the root section of the spit is exposed is on a southwesterly direction. The storm waves also come from this direction. This accounts for alignment of the root section of the spit, as storm waves tend to build shingle features that are aligned normal to their direction of approach.

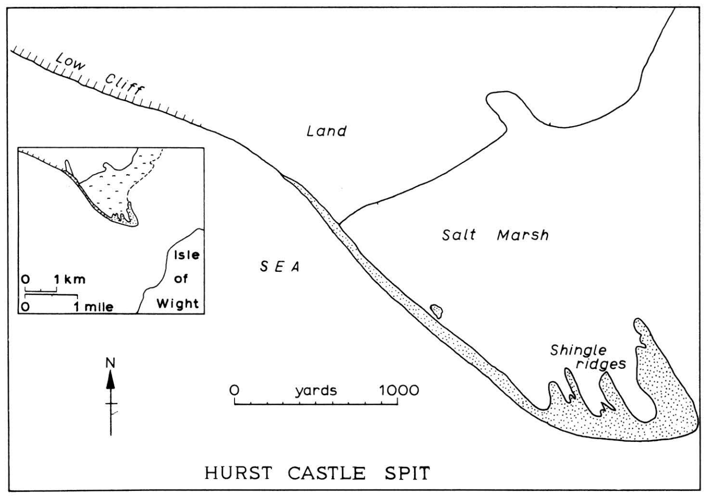


Figure 1.- Location of Hurst Castle spit.

The next largest waves, coming from the open Atlantic, are generated by the prevalent westerly winds and have a component that induces beach drifting along the spit from the west. The action of these waves, which are important in building the spit because they supply the bulk of material from which it is formed, also are simulated in the program. A storm by the action of the oblique waves throws material derived from the west up onto the back of the beach to form the storm ridge. At the same time they denude the foreshore by their destructive action. This material must be replaced before the spit can grow longer.

The major bend of large radius of curvature along the spit probably is due to the effect of wave refraction. As the spit extends away from the land it enters deeper water. The storm waves coming from the southwest, together with westerly waves, will suffer increasing refraction, progressively turning more to approach the shore from the south rather than southwest. Therefore, the effect of refraction also must be allowed in the model. This is achieved by

[1] Loose and waterworn flattened gravel and pebbles.

insertion of a mechanism that allows jumps to occur as the spit grows across the matrix.

Another important effect of the increase of depth offshore is the slowing of the easterly growth of the spit with time. As the water becomes deeper, more and more shingle is needed for each unit of growth. This factor also is allowed for in the program. Its main effect on the form of the spit is to increase the length and frequency of recurves towards the distal end of the spit. The real spit has few recurves along its proximal portion. This may be due partly to their burial in the salt marsh that is growing in the shelter of the spit, but also to the more rapid growth eastwards of the spit where the water is shallower near the coast.

The material forming the recurves must move along the spit and pass northwards around its sharp angle when waves come from a southerly or southeasterly direction. The fetch is short in these directions owing to the presence of the Isle of Wight. The waves would be short and little refracted. They therefore would be able to move the shingle along the spit in a northerly direction. Being fairly small the waves would not build large or permanent features, but could supply the material of which the recurves are formed. The sharp angle of the spit also is due to the Isle of Wight, which prevents large waves coming from the southeast.

The recurves face northeast along the Solent. The ridges are built by the waves generated by northeasterly winds blowing along the long stretch of open water in this direction. The recurves thus represent the action of waves coming from a direction almost diametrically opposite the storm waves that form the main spit ridge. This accounts for the small angle of 30° separating the alignment of the proximal part of the main ridge and the recurves. Thus four different types of waves are simulated in the program as well as the effects of refraction and increasing depth. The form of the spit depends essentially on these variables.

PROGRAM OPERATION

The development of the computer simulation model is based on the operation of fixed processes which follow one by one in a random fashion, although over the completely built spit there is a fixed proportion of operation of each process. The mainline program calculates random numbers in the range 0 to 100, and then matches the numbers with the fixed proportions of the total range allocated to each process routine. The split of the range might be for the WEST routine to be called if the random number falls between 0 and 40, STORM if between 41 and 55, SEPERP 56 to 73, and NOBLE 74 to 100. All values are variable as input and by variation of the numbers, the growth of the spit can be altered. The process routines WEST, STORM, SEPERP, and NOBLE are interactive in that the result of one will affect another, and all routines use a common indexing system.

To understand the marker system employed it is first necessary to consider what features the process routines will build. A typical small simulated spit in the early stages of growth and exhibiting all possible printing features for all the process routines is shown.

	1	2	3	4	5	6	7	8 - cols.
Row 45				7				
Row 46					9	7		
Row 47							3	
Row 48							1	
Row 49	2	2	2	1	1	1	1	1
Row 50	1	1	1	1	1			

The action of WEST is indicated by a 1, a completed STORM by a 2, a SEPERP by 1, and a NOBLE by 3, 7 and 9. Markers are needed in the program to store the row and column numbers of the ends of all features. For instance, a marker is needed to store the position of the 2 in column 3, as this column marks the point before which STORM will have no effect. Similarly the SEPERP action represented by a 1 in row 48 column 7 is noted, as is the end of the oblique recurve created by routine NOBLE which is marked with a 7 in row 45 column 4.

A previous simpler version of the program did not use as many markers to indicate the position of the features of the spit and CPU time was about 2 minutes. With a more complicated program using the marker system as much as possible, CPU time is only about 1 minute (of which compilation, organization and relocation take about 35 seconds). The program deck also is shorter.

PROGRAM DESCRIPTION

Main Program

A generalized flow diagram of the main program is given in Figure 2. The functions of this chain are to: (1) read the data; (2) empty the matrix in which the simulated spit is to be held; (3) set the markers necessary in operation of the program, and (4) print maps of the growing spit at specified intervals. The four process routines that can be used to make the spit grow are dealt with later. The arrays in use are:

(1) ICE(50,60) — which holds the simulated spit

(2) IONE(2) — stores row and column numbers of last element placed by the WEST routine, or after a STORM routine call, it may contain the last placed WEST

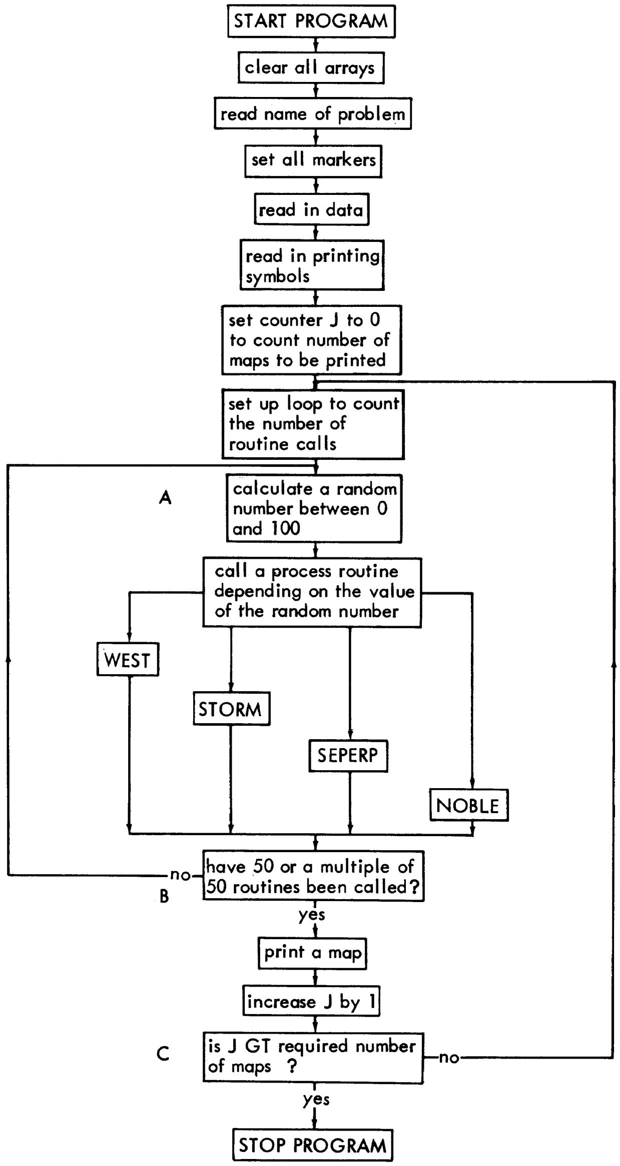


Figure 2.- Flow chart of main program.

not eroded by STORM

(3) ITWO(2) holds row and column numbers of the position in the growth of the spit to the left of which STORM will have no further effect

(4) JUMP(60,2) keeps a record of the row and column number prior to a successful jump (refraction) of the spit

(5) JURP(2) holds references of last placing of a successful action of SEPERP

(6) ISEV(2) contains references of next placing of a lateral recurve element; operative under NOBLE

(7) ITEST(60) maintains a record of whether a SEPERP has placed a 1 in the Storm row of a particular column

(8) MAP(10) holds alphanumeric characters for printing maps of the growth of the spit

Arrays 2 to 8 are set to appropriate values prior to the start of simulation.

The variables needed to delimit the action of the process routines are IA, IB, IC, ID and IE with values ranging from 0 to 100. These are read as data, as are the following variables

(1) K number of maps required

(2) IZAX starting number for the random sequence

(3) ADEPTH column number of limit of linear depth increase

(4) BDEPTH limit of growth of the spit specified by column number

(5) REF a figure indicating the speed of refraction required

In addition, there are some internally used COMMON variables

(6) M1 used for random number generation

(7) IPL indicates how many jumps have taken place

(8) LUMP at any point in time indicates column number of the West row at the end of the spit

(9) LEND marks highest column in use by the spit

(10) MARK keeps location of the top of the spit for printing purposes

(11) T stores floating point version of M1

Action of Routines

There are a total of six routines of which four are geomorphic process routines dealing with different aspects of the growth of the spit and two are administrative ones dealing with map printing and random number generation.

(1) RANDOM
This routine uses the power residue method to calculate a random number (N), which then is converted into an integer in the range supplied by the routine call.

(2) PMAT
The simulated spit is printed in A2 format using the characters supplied in array MAP. The program terminates if the argument of the routine is 0. Rows are numbered on the print to facilitate reference.

(3) WEST
The fundamental initial building of the spit is accomplished by this routine. It simulates the supply of shingle to the spit from the beach to the west and its action consists of placing a 1 in the simulation matrix ICE. Under normal conditions this extra material elongates the spit by placing a 1 on the bottom row at the left-hand side of the matrix. The effect of 3 normal calls of WEST is

```
          1  2  3  4  - cols.

Row 48
Row 49
Row 50    1  1  1
```

Further calls of the routine would progressively elongate the spit along this one row. The 'normal' conditions are, as usual, highly artificial - no allowance has been made for any increase in depth offshore, nor for the refraction effect. To simulate the change in depth two variables are used, ADEPTH and BDEPTH, which are set at the value of certain column numbers. With increasing water depth, elongation of the spit would slow and then eventually come to a halt. ADEPTH is designed to control the rate of elongation from column 0 to column ADEPTH. BDEPTH allows some elongation until, theoretically, the spit has reached the column number to which BDEPTH has been set. To the column number represented by ADEPTH, the decrease in growth is linear and gradual, but after ADEPTH and to BDEPTH the rate of seaward growth of the spit slows in a roughly exponential manner. To allow for this in the routine the column number where the 1 may be put is compared with a random number between 0 and BDEPTH. If the random number generated is greater than the column number and the column number is less than ADEPTH a 1 is placed in the matrix, otherwise another random number is generated between ADEPTH and BDEPTH. If the random number now is greater than the column number, the 1 is placed as usual, otherwise the control returns to the mainline program. This means that as the column number grows larger the spit itself enlarges so there is progressively less chance of placing a 1 to indicate the extension of the spit. Although there are three 1s in the above example, this may be the result of more than 3 calls of WEST. In this particular situation, however, it is unlikely.

Refraction is a similar problem in that greater refraction is required at the end of the spit in a roughly exponential manner. There is not only a choice between placing a 1 or not, but also in whether refraction is to take place. If a 1 is to be placed, the possibility of refraction is tested by comparing the square of the column number where placement would occur with a random number generated in the range 0 to REF. REF is placed conveniently around 2300 to best fit Hurst Castle spit. If the random number is less than the square of the column number, a jump (refraction) will take place unless the place where the 1 will be put is not in fact at the end of the spit. This can happen where the spit growth has been affected by STORM. Illustration of the situation shows the normal effect of a jump - that of putting a 1 on the row above West row in the same column as would have been used for a normal WEST.

```
          1  2  3  4  - cols.

Row 48
Row 49             1
Row 50    1  1  1
```

The effect of another WEST is shown by placing a 1 on the new West row.

```
          1  2  3  4  5  - cols.

Row 48
Row 49             1  1
Row 50    1  1  1
```

Jump cannot be effective because of a previous STORM.

```
          1  2  3  4  5  - cols.

Row 48                1  1
Row 49       1  1  1
Row 50    1
```

Note that in the second example a subsequent call of WEST will place a 1 in the new West row, in this situation row 49. From this stage on all growth in the spit will treat row 49 as the bottom row of the matrix. If a further jump were to occur, row 48 then would be the West row for all placing purposes to the right of the jump column. The last example shows how a jump from row 50 to row 49 is not possible if a STORM has taken place, because some of the 1s on row 50 have yet to be replaced after washed away by STORM. Thus only a normal call of WEST as in the first example can rebuild the spit until the point is reached where the West row (up to column 3 on row 50, then on row 49 for subsequent columns) is once again continuous where the previous example now has been filled in by the operation of WEST.

	1	2	3	4	5 - cols.
Row 48				1	1
Row 49	1	1	1	1	1
Row 50	1	1	1		

The operation of WEST is outlined in a generalized flow chart (see the Appendix).

(4) STORM

This routine is concerned with the effect of storms on the spit. In reality, these have the effect of eroding the seaward face of the spit where the shingle from the west has accreted. At the same time a storm ridge is built up landward behind the line of westerly accretion. This action has been simulated in routine STORM in the following manner. Where a spit has developed by a succession of calls of WEST the STORM routine removes the West row and adds a 1 to the Storm row immediately behind the West row, at the same time setting the West row to 0. This happens on both untouched and refracted sections of spit. The spit may be built up of a series of Wests with one jump.

	1	2	3	4	5	6 - cols.
Row 48						
Row 49				1	1	1
Row 50	1	1	1			

After a STORM has operated, the result is as follows.

	1	2	3	4	5	6 - cols.
Row 48				1	1	1
Row 49	1	1	1			
Row 50						

The Storm row in this example is row 49 to the left of the jump and row 48 to the right of it. If, at the stage shown in the latter diagram, a further STORM was called nothing would alter. Should more WESTs be called, however, the West row can fill in to produce the spit as shown after four successful calls of WEST.

	1	2	3	4	5	6 - cols.
Row 48				1	1	1
Row 49	1	1	1	1		
Row 50	1	1	1			

A STORM would alter the result so the 1s of the Storm row have been changed to 2s and the 1s of the West row have been zeroed.

	1	2	3	4	5	6 - cols.
Row 48				2	1	1
Row 49	2	2	2			
Row 50						

Although columns 1 to 4 have been changed to 2s in the Storm row, columns 5 and 6 have not as there was nothing in the West row beneath them. After five more successful WESTs the following situation is reached.

	1	2	3	4	5	6 - cols.
Row 48				2	1	1
Row 49	2	2	2	1	1	
Row 50	1	1	1			

The presence of a 2 indicates that the storm ridge has attained its highest possible level and thus cannot be built up further, thus the West row 1s are not eroded in columns 1 to 4 inclusive but remain untouched. As shown previously though, column 5 produces a 2 in the Storm row and column 6 remains unchanged. The result is shown after another STORM.

	1	2	3	4	5	6 - cols.
Row 48				2	2	1
Row 49	2	2	2	1		
Row 50	1	1	1			

It can be seen that STORM affects the spit directly on the availability of WESTs; it cannot operate unless there is some material in the West row. At any single column of the simulation matrix it is possible to define a West row and a Storm row, although in some situations one or the other of them will be empty. The position of the West row is always at the [(50 - (number of jumps to that point)] row, and the Storm row is always the row immediately behind. As long as the West row for a particular column is known, all other calculations can be made with reference to it. Thus the problem of marking the ends of specific features of the simulated spit for indexing purposes is simplified.

A flowchart of the STORM routine is given in the Appendix. The cycle from label 102 to label 100 in the flowchart delineates the control for searching each level of the West row of the spit. For the last diagram it would be necessary to go around the loop twice, once for every West row movement, i.e., the first loop from row 50 in column 1 to 3 and then in row 49 from columns 4 to 6. It is obvious that it is not necessary to proceed further than column 5 because there is nothing in the West row thereafter and any parts of the spit above the West row would not be affected. The first instruction after label 102 in the flowchart puts this into practice. Similarly, there can be no possible effect by STORM before column 5. Hence there is a marker which would be set to column 4 to indicate that no STORM action is possible before that point. This instruction is found in the second instruction after label 104 on the flowchart. The first instruction after 104 is to allow for the eventuality of a SEPERP's 1 being placed in the Storm row rather than the result of a STORM.

(5) SEPERP

This routine takes into account the effect of southeast winds producing a perpendicular recurve on the tip of the spit. Irrespective of whether the tip of the spit is formed by WEST or STORM, or the beginning of a recurve by NOBLE, this routine will place a 1 on the row immediately above the top row used in the last column of the simulated spit. Thus if before SEPERP the situation is

	1	2	3	4	5 - cols.
Row 48					
Row 49	2	2	2	1	1
Row 50	1	1	1		

the result of two SEPERPs placed is

	1	2	3	4	5 - cols.
Row 47					1
Row 48					1
Row 49	2	2	2	1	1
Row 50	1	1	1		

In reality the proportion of southeast winds that are effective in building the spit falls off towards the tip, largely replaced by northeast winds forming oblique recurves. To allow for this in the routine an exponential decay factor has been included at the beginning of the routine to determine whether there will be any SEPERP action as a result of this particular routine call. It is assumed that there will be no growth of SEPERPs beyond the 50 column mark because this point roughly corresponds to the spit chosen as an example. To vary the column beyond which no further SEPERP growth can be made is simply a matter of replacing the square of 50 (2500) by some other number. Once the column has been decided a random number can be calculated in the range 0 to 2500 (or other) and if it is less than or equal to the square of the column number in which the SEPERP's 1 would be placed, the routine returns control to the mainline program. One difficulty mentioned in routine STORM can arise. If, in a particular section of the matrix, many 'normal' WESTs have been placed successfully, the SEPERPs 1 is placed, as would be expected, at the end of the spit in the 49th row. This is, however, the Storm row and as SEPERP places a 1 this is indistinguishable from a 1 added to nothing on row 49 by a STORM. Before SEPERP the situation is

	1	2	3	4 - cols.
Row 49	2			
Row 50	1	1	1	1

The obvious alternative would be to give SEPERP a different value for placing purposes but in reality a new SEPERP probably would be turned into a storm ridge by a STORM. Thus, after SEPERP action, because material is present when a storm occurs, the resulting spit would have a higher distal end of the spit (on theStorm row) caused by a successful SEPERP.

	1	2	3	4 - cols.
Row 49	2			1
Row 50	1	1	1	1

The visible result of a STORM on a SEPERP is

	1	2	3	4 - cols.
Row 49	2	1	1	2
Row 50	1			

This result shows the presence of the SEPERP at the end of the spit. After rebuilding the West row, the effect of another STORM is to make the lone SEPERP indiscernable. This is not a drawback because it is impossible to tell in the real spit. After three successful WESTs, the result is

	1	2	3	4 - cols.
Row 49	2	1	1	2
Row 50	1	1	1	1

After STORM it would be

	1	2	3	4 - cols.
Row 49	2	2	2	2
Row 50	1			

Notice that under normal STORM conditions the 1 in column 4 of the former diagram would be left in place, but this would be logically impossible as:

(1) a storm acts on the whole spit and not just part of it,

(2) the WESTs would not be able to build past column 4 because of the blocking 1 in that column, and

(3) isolated patches of beach do not occur in reality.

Therefore the offending 1 is removed to produce the result shown in the last diagram. The instruction for this operation is labelled 109 in the flowchart.

In the situation of placing a SEPERP's 1 in a column where a SEPERP has been placed, the routine is simple in that the new 1 is placed immediately above the previous SEPERP. This is instruction 101 as shown in the flowchart. The 'STOP PROGRAM' instruction in the diagram is to terminate all running of the program in the situation of the SEPERP trying to build out of the top of the matrix.

(6) NOBLE

This routine stands for 'north - oblique - east'; in other words, for the action by winds from the northeast forming a recurve oblique to the principal line of spit growth. A typical spit end development just before the action of a NOBLE call is shown in the next diagram. There has been as yet no jump in the figure so that the 1 in column 5 is the direct result of the STORM sequence.

	1	2	3	4	5 - cols.
Row 48					
Row 49	2	1	1	1	1
Row 50	1	1			

The action of NOBLE, as outlined in the flowchart (see the Appendix), is to place a 3 on the Storm row in the place of the 1 in the 5th column to indicate the takeoff point of the new recurve. The 3 then is followed by a 7 placed in the matrix hole one row and one column less than the position of the 3. This assumes that there is room to place the 7 without hitting a part of the spit already developed. The resultant spit after NOBLE is as follows.

	1	2	3	4	5 - cols.
Row 48				7	
Row 49	2	1	1	1	3
Row 50	1	1			

Development of the NOBLE recurve should not be at a 45° angle to the horizontal, as would happen only if growth of the type similar to 7 were used, but closer to 30°. To compensate for this in the further building of the spit by NOBLE a 7 is placed only if a random number calculated within the range 1 to 10 falls between 1 and 4 inclusive. If the random number is between 5 and 10, a 9 is placed in the column immediately next to the already placed 7 (or 9). The situation after three calls of NOBLE would be

	1	2	3	4	5 - cols.
Row 47	9	9	7		
Row 48				7	
Row 49	2	1	1	1	3
Row 50	1	1			

Any future 7s would be placed diagonally above the previous figure. Should an obstruction get in the path of the growth of the recurve, a new one is automatically started. It is assumed that no progress can be made across a feature of the spit already present. Should the column number of a proposed extension go zero or negative the program again will start a new recurve.

If a SEPERP is placed in the row above the beginning of a NOBLE recurve the further growth of the oblique recurve is halted automatically and another new oblique started from the point where the SEPERP was placed.

Difficulties could arise if this were the only constraint on growth of oblique recurves. The other main problem is that a recurve could grow infinitely long if it originated in a column of the matrix from 50 onwards. This is because, with all SEPERP curtailed, no effective STORM action and no WEST placement, there is no possibility for the growth of new recurves, because the only operative process routine left is NOBLE. As long as the recurve does not hit a piece of the spit, an unlikely event at this stage because of the increased refraction at the tip of the spit, or deplete the matrix, it is likely to perpetuate itself indefinitely. It does seem reasonable, however, for this situation to develop as it would indicate the "old age" of the spit leading to the consolidation of the area now bounded triangularly by the land, storm ridge and long recurve. No further growth would be possible once the recurve had reached the land.

The routine is in two sections: (1) finding a place to put a number, and (2) actually placing it and resetting any necessary markers. The difference in calculating where to place a number and the actual placing of it is well displayed in the last diagram.

DATA INPUT

This is extremely simple and consists of 3 cards. The first card has any alphanumeric title in its 80 columns and the second holds 10 numbers in the FORMAT (6I5, I10, 3F10.3). The ten numbers consist of the following variables:

(1) minimum value for WEST operation

(2) maximum value for WEST operation

(3) maximum value for STORM operation

(4) maximum value for SEPERP operation

(5) maximum value for NOBLE operation

(6) number of maps to be printed, e.g. 10

(7) a starting random number, e.g. 23458913

(8) a terminator for linear depth increase

(9) a terminator for exponential depth increase

(10) a value to control spit refraction

The third card, punched in FORMAT (6A2), must contain the print symbols for the routines in the following order:

(1) symbol for open sea

(2) symbol for WEST

(3) symbol for STORM

(4) symbol for base of oblique recurve (NOBLE)

(5) symbol for arm of recurve (NOBLE) for least frequent printing

(6) symbol for arm of recurve (NOBLE) for most frequent printing

Typical punchings for the three cards could be:

(card 1) THIS IS A TYPICAL SET OF TEST DATA.

(card 2) 00000000400005500073001000001O
0023458913000040.000000058.000
003600.000

(card 3) ⎵⎵⎵ 1 ⎵ 2 ⎵ 3 ⎵ 7⎵ 9
(where a ⎵ indicates a blank column on a card)

EXAMPLE

The example of the Hurst Castle spit used here has similar data input requirements to those given in the previous section. Card one is punched with the title:

(card 1) A STANDARD SIMULATION RUN OF HURST CASTLE SPIT.

The second data card has the following parameters:
(card 2) 000000006500080000880010000015
0023458913000035.000000055.000
002300.000

Figure 3 shows the output at two stages in the run indicating the position of the spit immediately after map 5 and map 15. A comparison of the real spit in Figure 1 and the simulated spit (Fig. 3) shows a high degree of similarity.

CONCLUSION

The results of the simulation model indicate that the processes simulated probably are those important in the formation of Hurst Castle spit because the computer-generated spit has a close resemblance to the essential elements of the real spit. The refraction factor introduced in routine WEST produces a curvature that closely resembles the true spit, thus suggesting that this may be an important natural factor. The depth factor also has the effect of increasing the length and number of recurves in a realistic manner. A computer-simulation model of this type allows rapid experimentation with a variety of different values of the frequency of occurrence and intensity of the different processes. Thus the refraction factor and the depth factor can be changed to study the effect of changes and the optimum value can be selected, giving some indication of the natural value of these two variables. The frequency of occurrence of the different wave processes also readily can be changed, and again the optimum combination can be selected from the computer results. The standard computer run provides a close approximation to the outline of the real spit, confirming the importance of the different waves in forming the pattern of the spit and its recurves and suggesting that their relative frequency is of the same order as that in the program.

One of the main advantages of this type of computer simulation program is the speed with which different controls can be changed and in this it is much more rapid and inexpensive than a scale model, once the program has been written. Another advantage is the ability to simplify the real situation and test variations in the simplified process operations.

REFERENCES

Lewis, W.V., 1931, The effect of wave incidence on the configuration of a shingle beach: Geog. Jour., v. 78, no. 2, p. 129-148.

Lewis, W.V., 1938, The evolution of shoreline curves: Proc. Geol. Assoc., v. 49, no. 2, p. 107-127.

Listing of computer program

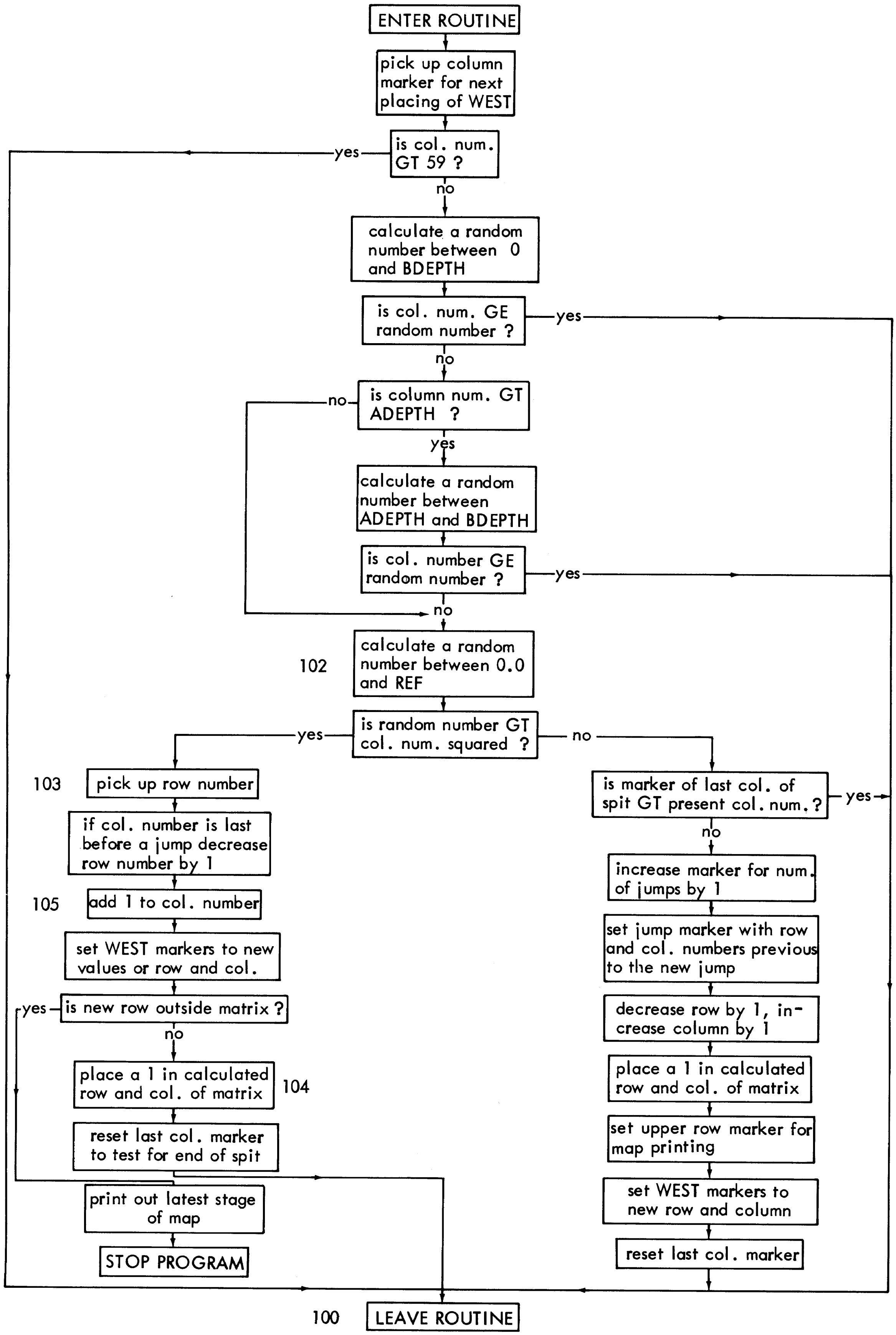
```
*FORTRAN
      COMMON ICE(50,60),IONE(2),ITWO(2),JUMP(60,2),JURP(2),ISEV(2),
     1ITEST(60),MAP(10),M1,IZAX,IPL,LUMP,LEND,ADEPTH,BDEPTH,REF,MARK,T
     2,JDEPTH
      DIMENSION NAME(20)
      M1=27548923
      DO 110 I=1,60
  110 ITEST(I)=0
      T=FLOAT(M1)
C      CLEAR SIMULATION MATRIX
      DO 100 I=1,50
      DO 100 J=1,60
  100 ICE(I,J)=0
      READ(5,202)(NAME(I),I=1,20)
      WRITE(6,202)(NAME(I),I=1,20)
C      SET MARKERS
      MARK=50
      IONE(2)=0
      IONE(1)=50
      ITWO(1)=49
      ITWO(2)=0
      IPL=0
      JUMP(1,1)=50
      JUMP(1,2)=60
      JURP(1)=50
      JURP(2)=1
      ISEV(2)=0
      LUMP=50
      LEND=0
C      DATA INPUT IS AS FOLLOWS,
C      MIN. VALUE FOR WEST
C      MAX FOR WEST
C      MAX FOR STORM
C      MAX FOR SEPERP
C      MAX FOR NOBLE
C      NO. OF MAPS REQUIRED
C      STARTING RANDOM NUMBER, 9 DIGITS
C      TERMINATOR FOR LINEAR DEPTH INCREASE
C      TERMINATOR FOR EXP. DEPTH INCREASE
C      REFRACTION CONTROL VALUE
C      6 MAP PRINTING SYMBOLS
      READ(5,200)IA,IB,IC,ID,IE,K,IZAX,ADEPTH,BDEPTH,REF
      WRITE(6,201)IA,IB,IC,ID,IE,K,IZAX,ADEPTH,BDEPTH,REF
      READ(5,203)MAP(1),MAP(2),MAP(3),MAP(4),MAP(8),MAP(10)
      I=0
      J=0
      JDEPTH=INT(ADEPTH)
  102 DO 108 I=1,50
  103 CALL  RANDOM(N,0.0,100.0)
C      CHOOSE PROCESS ROUTINE
      IF((N.GT.IA).AND.(N.LE.IB))GOTO104
      IF(N.LE.IC)GOTO105
      IF(N.LE.ID)GOTO106
      IF(N.LE.IE)GOTO107
      GO  TO  103
  104 CALL  WEST
      GO  TO  108
  105 CALL  STORM
      GO  TO  108
  106 CALL  SEPERP
      GO  TO  108
```

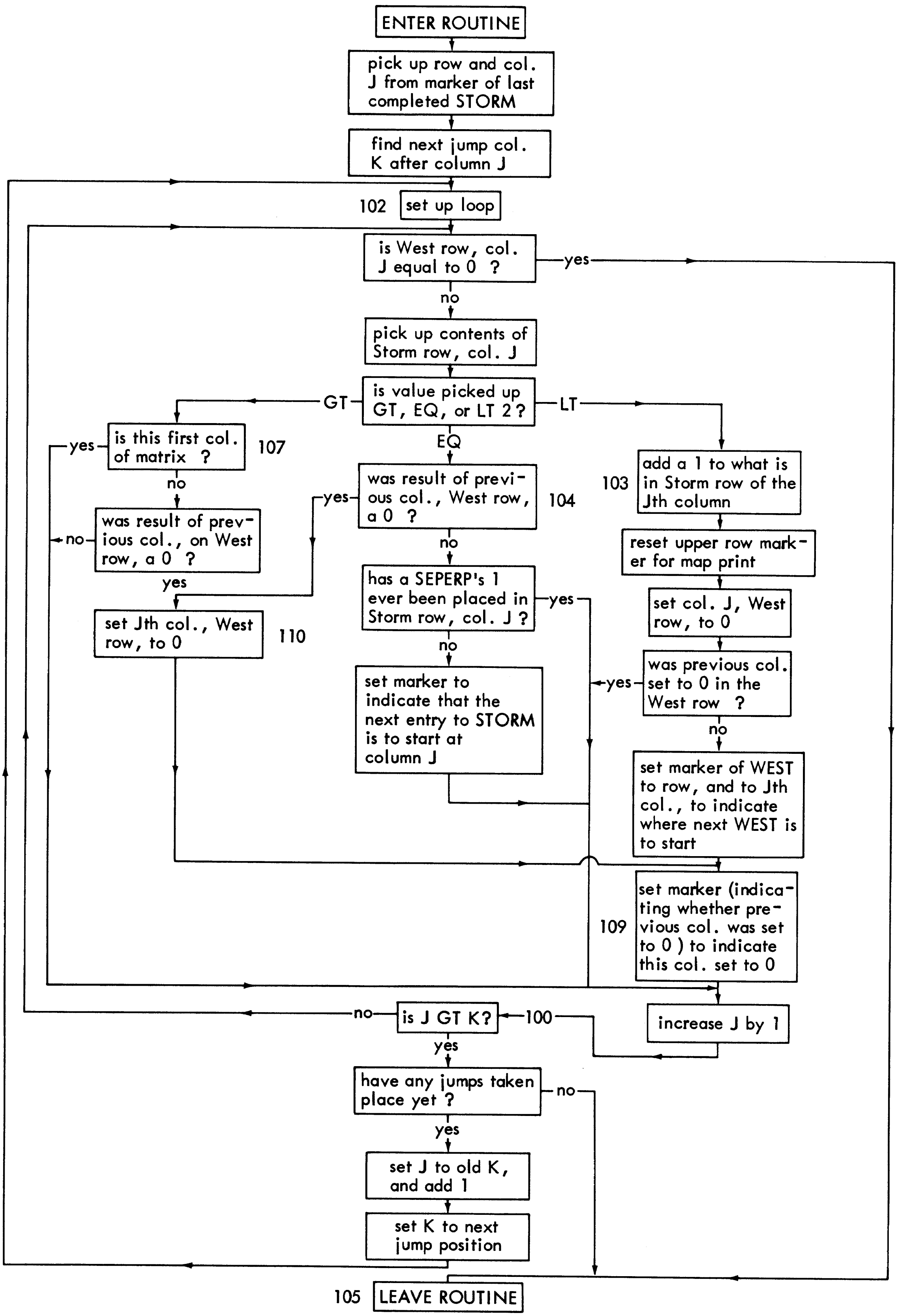
```
  107 CALL   NOBLE
  108 CONTINUE
      CALL   PMAT(1)
      J=J+1
      IF(J.EQ.K)GOTO109
      GO  TO  102
  109 CALL  EXIT
  200 FORMAT(6I5,I10,3F10.3)
  201 FORMAT(1H0,7I10,3F10.3)
  202 FORMAT(20A4)
  203 FORMAT(6A2)
      END
*FORTRAN
      SUBROUTINE  RANDOM(N,B,C)
      COMMON ICE(50,60),IONE(2),ITWO(2),JUMP(60,2),JURP(2),ISEV(2),
     1ITEST(60),MAP(10),M1,IZAX,IPL,LUMP,LEND,ADEPTH,BDEPTH,REF,MARK,T
     2,JDEPTH
      IZAX=IZAX*101
      U=FLOAT(IZAX)/T
      I=INT(U)
      IF(U.LE.0.0)I=I+1
      ZZ=ABS(U-FLOAT(I))
      IZAX=INT(ZZ*T)
      N=INT(ZZ*(C-B)+B)
      RETURN
      END
*FORTRAN
      SUBROUTINE  PMAT(I)
      COMMON ICE(50,60),IONE(2),ITWO(2),JUMP(60,2),JURP(2),ISEV(2),
     1ITEST(60),MAP(10),M1,IZAX,IPL,LUMP,LEND,ADEPTH,BDEPTH,REF,MARK,T
     2,JDEPTH
      DIMENSION KAR(60)
      WRITE(6,200)
      DO 100 J=1,50
      IF(J.LT.MARK)GOTO 103
C      CHOOSE APPROPRIATE PRINT CHARACTER
      DO 102 K=1,LEND
      L=ICE(J,K)+1
  102 KAR(K)=MAP(L)
C      PRINT A LINE OF MATRIX IN A2
      WRITE(6,202)J,(KAR(K),K=1,LEND)
      GOTO 100
  103 WRITE(6,202)J
  100 CONTINUE
      IF(I.EQ.0)GOTO101
      RETURN
  101 CALL  EXIT
  200 FORMAT(1H1,26HSIMULATED  SPIT  FORMATION///)
  202 FORMAT(3HROW,I3,2X,60A2)
      RETURN
      END
*FORTRAN
      SUBROUTINE  NOBLE
      COMMON ICE(50,60),IONE(2),ITWO(2),JUMP(60,2),JURP(2),ISEV(2),
     1ITEST(60),MAP(10),M1,IZAX,IPL,LUMP,LEND,ADEPTH,BDEPTH,REF,MARK,T
     2,JDEPTH
      IF(LEND.EQ.0)GOTO100
      IZ=7
      KZ=0
C      IS THERE A SEPERP OR NOBLE AT
C      END OF THE SPIT
      IF(LEND.GT.JURP(2))GOTO101
      I=ISEV(1)
      K=ISEV(2)
```

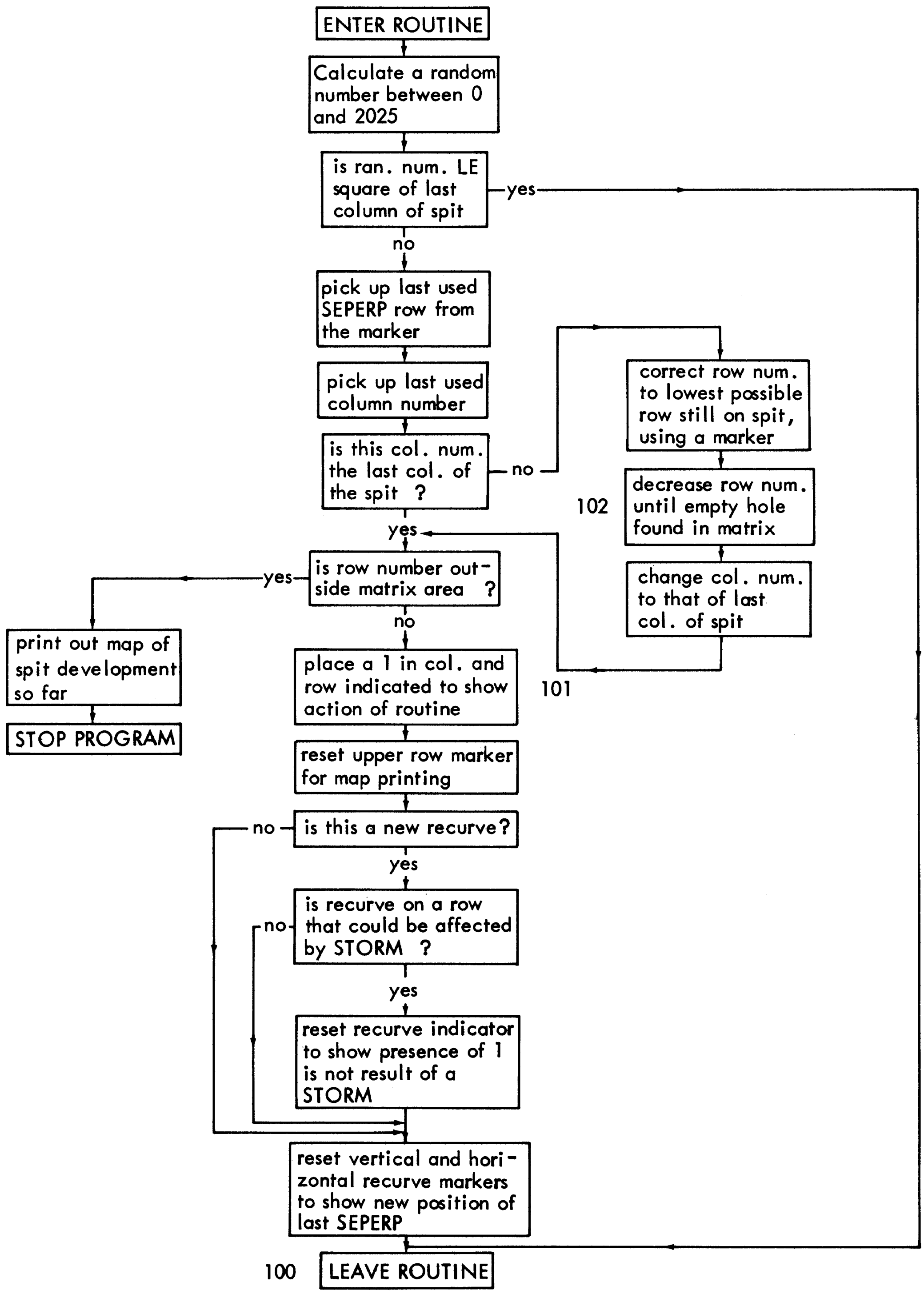
```
C        IS IT IMPOSSIBLE TO PLACE A 7 OR 9
       IF(K.EQ.0)GOTO104
       IA=I+1
       KA=K+1
       IF(ICE(IA,KA).LT.3)ICE(IA,KA)=3
       IF(ICE(IA,KA).EQ.3)KZ=1
C        IS RECURVE GOING OUT OF MATRIX
  102  IF((I.LT.1).OR.(K.LT.1))GOTO100
C        HAS A 3 BEEN PLACED THIS TIME
       IF(KZ.EQ.1)GOTO107
       CALL RANDOM(N,1.0,10.0)
       IF(N.LT.5)GOTO107
       I=I+1
       IZ=9
C        IS FURTHER GROWTH POSSIBLE WITHOUT CROSSING SPIT
  107  IF(ICE(I,K).GT.0)GOTO103
       ICE(I,K)=IZ
       IF(I.LT.MARK)MARK=I
       ISEV(1)=I-1
       ISEV(2)=K-1
       GO  TO  100
C        PICK UP PRESENT SEPERP ROW NUMBER
  104  I=JURP(1)
C        DECREASE ROW NUMBER UNTIL EMPTY ROW OF LAST
C        COLUMN FOUND
  105  I=I-1
       IF(ICE(I,LEND).NE.0)GOTO105
       GOTO 106
C        CALCULATE ROW NUMBER OF START OF NEW RECURVE
  101  I=LUMP-1
C        PUT A 3 IN CALCULATED ROW OF LAST COLUMN
  106  ICE(I,LEND)=3
       JURP(1)=I
       I=I-1
       K=LEND-1
       JURP(2)=LEND
       KZ=1
       GO  TO  102
C        SET MARKER TO INDICATE NO POSSIBILITY OF GROWTH
  103  ISEV(2)=0
  100  RETURN
       END
*FORTRAN
       SUBROUTINE  WEST
       COMMON ICE(50,60),IONE(2),ITWO(2),JUMP(60,2),JURP(2),ISEV(2),
      1ITEST(60),MAP(10),M1,IZAX,IPL,LUMP,LEND,ADEPTH,BDEPTH,REF,MARK,T
      2,JDEPTH
       I=IONE(2)
       IF(I.GT.59)GOTO100
C        NEXT 4 CARDS TEST IF WEST TO BE
C        ALLOWED TO OPERATE
       CALL  RANDOM(N,0.0,BDEPTH)
       IF(I.GE.N)GOTO100
       IF(I.LE.JDEPTH)GOTO102
       CALL RANDOM(N,ADEPTH,BDEPTH)
       IF(I.GE.N)GOTO100
C        IS A JUMP TO TAKE PLACE
  102  CALL RANDOM(N,0.0,REF)
C        JUMP SECTION OF WEST
       J=I*I
       IF(N.GT.J)GOTO103
        IF(I.EQ.0)GOTO100
C        IS MARKER OF LAST COL GT PRESENT COL
       IF(LEND.GT.I)GOTO100
```

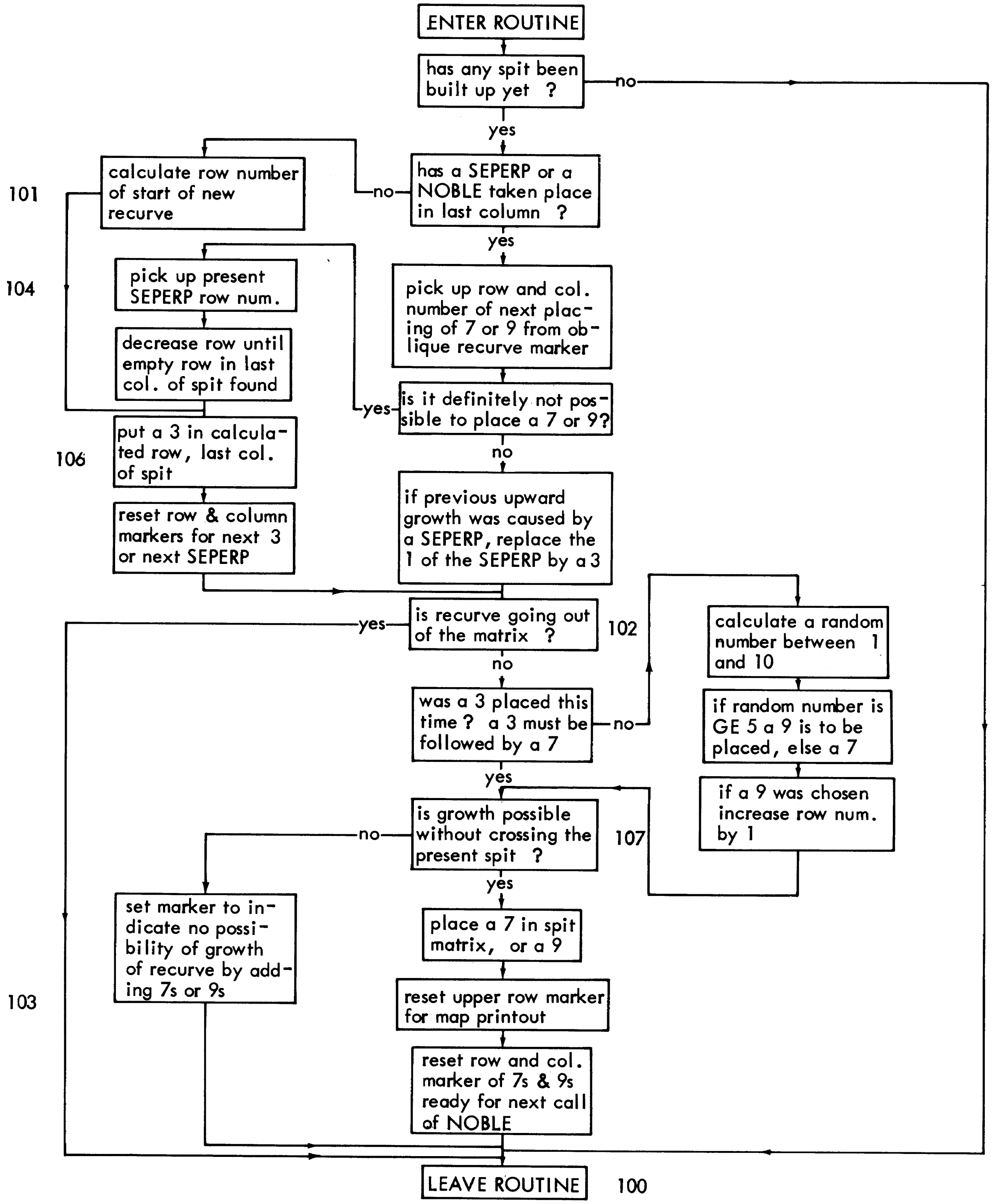
```
      K=I
C       SET MARKERS
      IPL=IPL+1
      JUMP(IPL,1)=LUMP
      JUMP(IPL,2)=K
      LUMP=LUMP-1
      IE=IPL+1
      JUMP(IE,1)=LUMP
      JUMP(IE,2)=60
      JURP(1)=LUMP
      K=K+1
      ICE(LUMP,K)=1
C       RESET UPPER ROW PRINTING MARKER
      IF(LUMP.LT.MARK)MARK=LUMP
      IONE(1)=LUMP
      IONE(2)=K
      LEND=K
      GO TO 100
C       NORMAL ACTION SECTION OF WEST
  103 K=IONE(1)
      J=50-K+1
      IF(I.NE.JUMP(J,2))GOTO105
      K=K-1
  105 I=I+1
      IONE(1)=K
      IONE(2)=I
      IF(K.NE.0)GOTO104
      CALL PMAT(O)
  104 ICE(K,I)=1
C       INCREASE THE END OF SPIT MARKER
C       IF NECESSARY
      IF(I.GT.LEND)LEND=I
  100 RETURN
      END
*FORTRAN
      SUBROUTINE  STORM
      COMMON ICE(50,60),IONE(2),ITWO(2),JUMP(60,2),JURP(2),ISEV(2),
     1ITEST(60),MAP(10),M1,IZAX,IPL,LUMP,LEND,ADEPTH,BDEPTH,REF,MARK,I
     2,JDEPTH
      I=ITWO(1)
      IA=50-I
      J=ITWO(2)+1
      K=JUMP(IA,2)
      IF(J.GT.K)J=K
      IB=I+1
      IF(K.GT.LEND)K=LEND
       IF(K.EQ.0)GOTO105
      JTEST=1
C       THIS CYCLE VARIES ACCORDING TO THE NUMBER
C       OF JUMPS THAT HAVE TAKEN PLACE
  102 DO 100  IC=J,K
      IF(ICE(IB,IC).EQ.0)GOTO105
      IE=ICE(I,IC)
      IF(IE-2)103,104,107
C       IS THIS FIRST COLUMN OF MATRIX
  107 IF(IC.EQ.1)GOTO100
      IF(JTEST.EQ.1)GOTO100
C       SET THE WEST ROW OF COLUMN TO 0
  110 ICE(IB,IC)=0
      GOTO 109
C       WAS RESULT OF PREVIOUS COLUMN, WEST ROW, 0
  104 IF(JTEST.EQ.0)GOTO110
      IF(ITEST(IC).EQ.1)GOTO100
      ITWO(1)=I
```

```
      ITWO(2)=IC
      GOTO 100
C      ADD A 1 TO STORM ROW
  103 IE=IB-1
      ICE(IE,IC)=ICE(IE,IC)+1
      IF(IE.LT.MARK)MARK=IE
      ICE(IB,IC)=0
      IF(JTEST.EQ.0)GOTO100
      IE=IB
      IF(IE.EQ.50)GOTO101
      IF(J.EQ.1)GOTO101
      IF(IA.EQ.1)GOTO101
      IX=IA-1
      IF((IC.EQ.J).AND.(J.EQ.JUMP(IX,2)+1))IE=IB+1
  101 IONE(1)=IE
      IONE(2)=IC-1
C      SET MARKER TO SHOW  THIS COLUMN ON WEST ROW
C      IS SET TO 0
  109 JTEST=0
  100 CONTINUE
C      SET UP NEXT JUMP LOOP
      IF(IPL.EQ.0)GOTO105
      J=K+1
      IF(J.GT.60)GOTO105
      IA=IA+1
      K=JUMP(IA,2)
      I=I-1
      IB=I+1
      GOTO 102
  105 RETURN
      END
*FORTRAN
      SUBROUTINE  SEPERP
      COMMON ICE(50,60),IONE(2),ITWO(2),JUMP(60,2),JURP(2),ISEV(2),
     1ITEST(60),MAP(10),M1,IZAX,IPL,LUMP,LEND,ADEPTH,BDEPTH,REF,MARK,T
     2,JDEPT''
      CALL  RANDOM(N,0.0,2025.0)
      I=LEND
      IF(I.EQ.0)I=1
      J=I*I
      IF(J.GT.N)GOTO100
      K=JURP(1)
      IF(I.GT.JURP(2))K=LUMP
C      DECREASE ROW NUMBER UNTIL EMPTY HOLE FOUND
  102 K=K-1
      IF(ICE(K,I).NE.0)GOTO102
      IF(K.GT.0)GOTO101
      CALL PMAT(0)
C      PLACE 1 IN COLUMN AND ROW INDICATED TO SHOW
C      SUCCESSFUL ACTION OF SEPERP
  101 ICE(K,I)=1
C      SET MARKER IF SEPERP IN STORM ROW
      IF(K.LT.MARK)MARK=K
      IE=LUMP-1
      IF(K.EQ.IE)ITEST(I)=1
      JURP(1)=K
      JURP(2)=I
      ISEV(1)=K-1
      ISEV(2)=I-1
  100 RETURN
      END
```


ENTER ROUTINE
pick up column marker for next placing of WEST
is col. num. GT 59 ?
yes
no
calculate a random number between 0 and BDEPTH
is col. num. GE random number ?
yes
no
is column num. GT ADEPTH ?
no
yes
calculate a random number between ADEPTH and BDEPTH
is col. number GE random number ?
yes
no
102
calculate a random number between 0.0 and REF
is random number GT col. num. squared ?
yes
no
103
pick up row number
if col. number is last before a jump decrease row number by 1
105
add 1 to col. number
set WEST markers to new values or row and col.
yes
is new row outside matrix ?
no
place a 1 in calculated row and col. of matrix
104
reset last col. marker to test for end of spit
print out latest stage of map
STOP PROGRAM
is marker of last col. of spit GT present col. num. ?
yes
no
increase marker for num. of jumps by 1
set jump marker with row and col. numbers previous to the new jump
decrease row by 1, increase column by 1
place a 1 in calculated row and col. of matrix
set upper row marker for map printing
set WEST markers to new row and column
reset last col. marker
100
LEAVE ROUTINE


ENTER ROUTINE
pick up row and col. J from marker of last completed STORM
find next jump col. K after column J
102
set up loop
is West row, col. J equal to 0 ?
yes
no
pick up contents of Storm row, col. J
is value picked up GT, EQ, or LT 2 ?
GT
LT
EQ
is this first col. of matrix ?
107
yes
no
was result of previous col., on West row, a 0 ?
no
yes
set Jth col., West row, to 0
110
was result of previous col., West row, a 0 ?
104
yes
no
has a SEPERP's 1 ever been placed in Storm row, col. J ?
yes
no
set marker to indicate that the next entry to STORM is to start at column J
add a 1 to what is in Storm row of the Jth column
103
reset upper row marker for map print
set col. J, West row, to 0
was previous col. set to 0 in the West row ?
yes
no
set marker of WEST to row, and to Jth col., to indicate where next WEST is to start
set marker (indicating whether previous col. was set to 0) to indicate this col. set to 0
109
increase J by 1
is J GT K?
100
no
yes
have any jumps taken place yet ?
no
yes
set J to old K, and add 1
set K to next jump position
105
LEAVE ROUTINE


ENTER ROUTINE
Calculate a random number between 0 and 2025
is ran. num. LE square of last column of spit
yes
no
pick up last used SEPERP row from the marker
pick up last used column number
is this col. num. the last col. of the spit ?
no
yes
correct row num. to lowest possible row still on spit, using a marker
102
decrease row num. until empty hole found in matrix
change col. num. to that of last col. of spit
101
is row number outside matrix area ?
yes
no
print out map of spit development so far
STOP PROGRAM
place a 1 in col. and row indicated to show action of routine
reset upper row marker for map printing
is this a new recurve?
no
yes
is recurve on a row that could be affected by STORM ?
no
yes
reset recurve indicator to show presence of 1 is not result of a STORM
reset vertical and horizontal recurve markers to show new position of last SEPERP
100
LEAVE ROUTINE


ENTER ROUTINE
has any spit been built up yet ?
no
yes
has a SEPERP or a NOBLE taken place in last column ?
no
yes
101
calculate row number of start of new recurve
104
pick up present SEPERP row num.
decrease row until empty row in last col. of spit found
106
put a 3 in calculated row, last col. of spit
reset row & column markers for next 3 or next SEPERP
pick up row and col. number of next placing of 7 or 9 from oblique recurve marker
is it definitely not possible to place a 7 or 9?
yes
no
if previous upward growth was caused by a SEPERP, replace the 1 of the SEPERP by a 3
is recurve going out of the matrix ?
yes
no
102
was a 3 placed this time ? a 3 must be followed by a 7
no
yes
calculate a random number between 1 and 10
if random number is GE 5 a 9 is to be placed, else a 7
if a 9 was chosen increase row num. by 1
107
is growth possible without crossing the present spit ?
no
yes
set marker to indicate no possibility of growth of recurve by adding 7s or 9s
103
place a 7 in spit matrix, or a 9
reset upper row marker for map printout
reset row and col. marker of 7s & 9s ready for next call of NOBLE
LEAVE ROUTINE
100

```
SIMULATED  SPIT  FORMATION

ROW  1
ROW  2
ROW  3
ROW  4
ROW  5
ROW  6
ROW  7
ROW  8
ROW  9
ROW 10
ROW 11
ROW 12
ROW 13
ROW 14
ROW 15
ROW 16
ROW 17
ROW 18
ROW 19
ROW 20
ROW 21
ROW 22
ROW 23
ROW 24
ROW 25
ROW 26
ROW 27
ROW 28
ROW 29
ROW 30
ROW 31
ROW 32
ROW 33
ROW 34
ROW 35
ROW 36
ROW 37
ROW 38
ROW 39                                                           9 7
ROW 40                                                               7
ROW 41                                                                 7
ROW 42                                             7                     7
ROW 43                                           9 7 3               9 7   3
ROW 44                                           9 7 3                   7 1 1
ROW 45                                             7 3                     3 3
ROW 46                   1                         7 3 7     1     2 2 2 2 2 1
ROW 47                 7 1                       2 2 3 3 3 2 2 2 2 1 1 1 1 1
ROW 48               7 7 3   1 7   2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1
ROW 49   2 2 2 2 2 2 2 3 3 2 2 2 3 1 1 1 1 1 1 1
ROW 50   1 1 1 1 1 1 1 1 1 1 1 1 1
```

```
SIMULATED  SPIT  FORMATION

ROW  1
ROW  2
ROW  3
ROW  4
ROW  5
ROW  6
ROW  7
ROW  8
ROW  9
ROW 10
ROW 11
ROW 12
ROW 13
ROW 14
ROW 15
ROW 16
ROW 17
ROW 18
ROW 19
ROW 20
ROW 21
ROW 22
ROW 23
ROW 24
ROW 25
ROW 26
ROW 27
ROW 28
ROW 29                                                                         9 7
ROW 30                                                                             9 7
ROW 31                                                                       7         7
ROW 32                                                                         9 7       7
ROW 33                                                                             7       7 7   7
ROW 34                                                                               7       7 9 7 3
ROW 35                                                                                 7 7 7 7 3 7 3
ROW 36                                                                                   7 7 7 3 7 3
ROW 37                                                                                     7 7 3   3
ROW 38                                                                                       7 3 1
ROW 39                                                           9 7                           3
ROW 40                                                               7         7             2 1
ROW 41                                                                 7         7 7       2 1
ROW 42                                             7                     7     9 7 3 7   2 1
ROW 43                                           9 7 3               9 7   3       3 3 3 1
ROW 44                                           9 7 3                   7 1 3   2 2 1 1
ROW 45                                             7 3                     3 3 3 1 1
ROW 46                   1                         7 3 7     1     2 2 2 2 2 1 1
ROW 47                 7 1                       2 2 3 3 3 2 2 2 2 1 1 1 1 1
ROW 48               7 7 3   1 7   2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1
ROW 49   2 2 2 2 2 2 2 3 3 2 2 2 3 1 1 1 1 1 1 1
ROW 50   1 1 1 1 1 1 1 1 1 1 1 1 1
TIME    18.05.29 RAN/EL/000M54S/001M56S

                ORDINAL NUMBER  03     FROM TAPE Y TSN OUT003

PRINTING COMPLETED
```

KANSAS GEOLOGICAL SURVEY COMPUTER PROGRAM
THE UNIVERSITY OF KANSAS, LAWRENCE

PROGRAM ABSTRACT

Title (If subroutine state in title):

SPITSYM, A FORTRAN IV COMPUTER PROGRAM FOR SPIT SIMULATION

Date: January 1970

Author, organization: M.J. McCullagh, Geography Dept., Nottingham University,
Nottingham, NG7 2RD., England

Direct inquiries to: author

Name: Address:

Purpose/description: To simulate the growth of a shingle spit given the processes forming the spit.

Mathematical method: Stochastic

Restrictions, range: Application only to this type of spit unless program altered.

Computer manufacturer: I.C.L. Model: KDF9, 32K, 48 bit words

Programming language: FORTRAN IV

Memory required: 6 K Approximate running time: 1 min CPU

Special peripheral equipment required:

Remarks (special compilers or operating systems, required word lengths, number of successful runs, other machine versions, additional information useful for operation or modification of program)

Originally written in Egdon FORTRAN, using 48 bit word length; should be satisfactory on 6 or 8 bit as no numbers dealt with are larger than 60, except for the 2 variable names M1, IZAX.

Some Recent Publications of Interest

COMPUTER CONTRIBUTIONS

40.	Symposium on computer applications in petroleum exploration, D.F. Merriam, editor, 1969	$1.00
41.	FORTRAN IV program for sample normality tests, by D.A. Preston, 1970	$1.00
42.	CORFAN-FORTRAN IV computer program for correlation, factor analysis (R- and Q-mode) and varimax rotation, by C.W. Ondrick and G.S. Srivastava, 1970 .	$1.50
43.	Minimum entropy criterion for analytic rotation, by R.B. McCammon, 1970	$1.00
44.	FORTRAN IV CDC 6400 computer program for constructing isometric diagrams, by W.B. Wray, Jr., 1970	$1.00
45.	An APL language computer program for use in electron microprobe analysis, by D.G.W. Smith and M.C. Tomlinson, 1970	$1.00
46.	FORTRAN IV program for Q-mode cluster analysis on distance function with printed dendrogram, by J.M. Parks, 1970	$1.00
47.	FORTRAN IV program for canonical variates analysis for the CDC 3600 computer, by R.A. Reyment and H. Ramden, 1970	$1.00
48.	The dendrograph, by R.B. McCammon and G. Wenninger, 1970	$1.00
49.	FORTRAN IV program for simulating geologic development of sedimentary basins, by D.R. Ojakangas, 1970	$1.00
50.	SPITSYM, a FORTRAN IV computer program for spit simulation, by M.J. McCullagh and C.A.M. King, 1970	$1.00

SPECIAL DISTRIBUTION PUBLICATIONS

38.	The unit regional-value concept and its application to Kansas, by J.C. Griffiths, 1969	$1.00
39.	FORTRAN IV program for synthesis and plotting of water-quality data, by L.H. Ropes, C.O. Morgan, and J.M. McNellis, 1969	$1.00
42.	FORTRAN IV program, KANS, for the conversion of general land office locations to latitude and longitude coordinates, by C.O. Morgan and J.M. McNellis, 1969 . .	$1.00
43.	Stiff diagrams of water-quality data programmed for the digital computer, by C.O. Morgan and J.M. McNellis, 1969	$1.00
45.	Modified piper diagrams by the digital computer, by J.M. McNellis and C.O. Morgan, 1969.	$1.00

REPRINTS (available upon request)

Mathematical conversion of section, township, and range notation to Cartesian coordinates, by D.I. Good (reprinted from Kansas Geological Survey Bulletin 170, pt. 3, 1964)

Digital computer methods for water-quality data, by C.O. Morgan, R.J. Dingman, and J.M. McNellis (reprinted from Ground Water, v. 4, no. 3, 1966)

Re-interpretation of the Cyrtina septosa Band data, Lower Carboniferous of Derbyshire, England, by computer analysis, by H.E. Sadler and D.F. Merriam (reprinted from Sedimentology, v. 8, no. 1, p. 55-61, 1967)

Computer use in Europe, by D.F. Merriam (reprinted from Geotimes, v. 12, no. 9, p. 14-16, 1967)

Development of computer applications to hydrology in Kansas, by C.O. Morgan, J.M. McNellis, and B.H. Lowell (preprint from Society of Petroleum Engineers of AIME, paper no. SPE 2080, 1968)