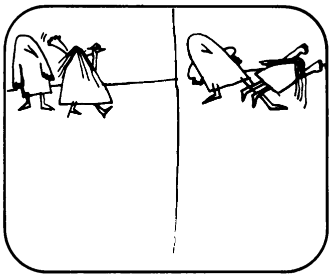
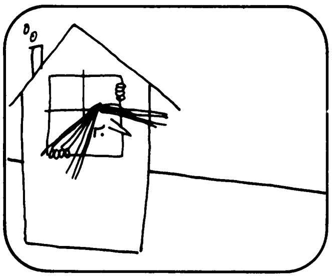
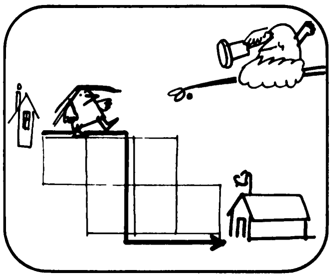
Perplexing Paths



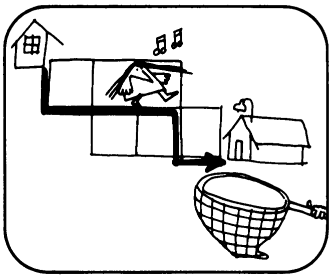
Susan has a problem. When she walks to school she keeps meeting Stinky.
Stinky: Hi Susan. Can I walk with you?
Susan: No Please go away.



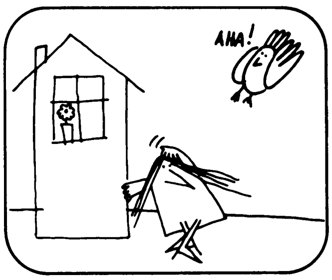
Susan: I know what I'll do. I'll walk to school a different way every morning. Then Stinky won't know where to find me.



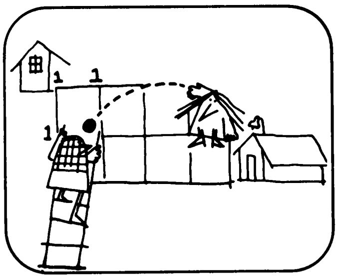
This map shows all the streets between Susan's house and her school. On this particular path Susan is always walking east or south



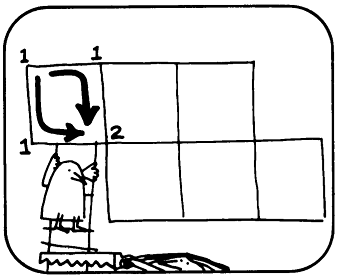
Here is Susan on another path. Naturally she doesn't want to walk away from the school But how many paths are there?



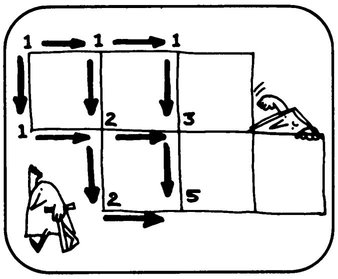
Susan: I wonder how many different ways I can go. Let's see. Hmm. This is going to be tough to figure out. Hmm. Aha! It's not hard at all It's simple! What insight did Susan have?



Here's how she reasoned.
Susan: I'll put a 1 at the corner where I live because I have just one way to start. Then I'll put a 1 at each corner that's one block away because there's only one way to get there



Susan: Now I'll put a 2 at this corner because I can get to it in two different ways.
When Susan noticed that 2 is the sum of 1 and 1, she suddenly realized that the number on every corner must be the sum of the one or two nearest numbers along the paths leading to that corner



Susan: There Four more corners are labeled. I'll soon finish the others.
Can you complete the labelling of the corners for Susan and tell her how many different ways she can walk to school?

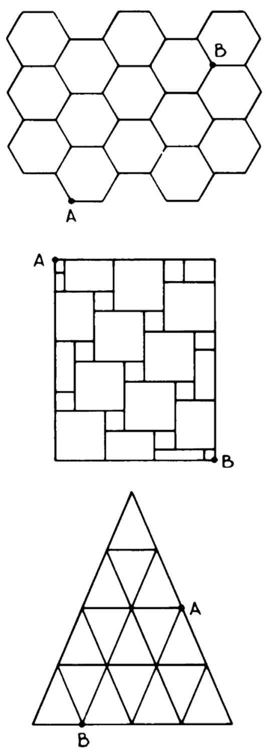
How Many Paths?

The remaining five vertices, reading top-down and left-right, are labeled 1, 4, 9, 4 and 13. The 13 at the last vertex shows that Susan has 13 ways to walk to school along shortest paths.

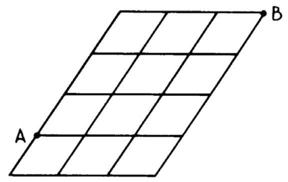
What Susan discovered is a simple, fast algorithm for calculating the number of shortest paths from her house to school. Had she attempted to draw all these paths, then count them, it would have been tedious, and out of the question if the street grid had contained a very large number of cells. You will better appreciate the algorithm's efficiency if you actually trace all 13 paths.

To test your understanding of the algorithm, try sketching a variety of other street networks and applying the algorithm to determine the number of shortest paths from any vertex A to any other vertex B. Figure 1 gives four problems of this type. They can be solved in other ways, using combinatorial formulas, but the methods are tricky and complicated.

1



1 (continued)



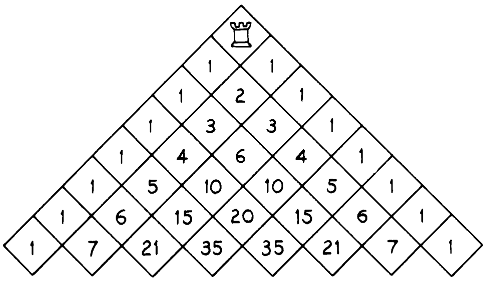
What is the number of shortest paths by which a chess rook can move from one corner of a chessboard to the diagonally opposite corner? This problem is quickly solved by labeling all the cells of the board in the same manner that Susan labeled the street corners. A chess rook moves only along orthogonals (horizontally and vertically), therefore the shortest paths are obtained by confining each move to a direction that carries the rook toward its goal. When the entire board has been labeled correctly, as shown in Figure 2, the labels will give at once the number of shortest paths from the starting square to any square on the board. The cell at the upper right corner has the number 3,432, therefore there are 3,432 ways that the rook can go from one corner to the diagonally opposite corner along shortest routes.

2

1	8	36	120	330	792	1716	3432
1	7	28	84	210	462	924	1716
1	6	21	56	126	252	462	792
1	5	15	35	70	126	210	330
1	4	10	20	35	56	84	120
1	3	6	10	15	21	28	36
1	2	3	4	5	6	7	8
♜	1	1	1	1	1	1	1

Let us slice the chessboard in half along a diagonal, then turn it so it becomes the triangle shown in Figure 3. The numbers on the bottom row of cells give the number of shortest paths from the apex cell to each cell at the bottom. The labeling of this triangle is identical with the numbers of Pascal's famous number triangle. The algorithm for computing the shortest paths from the top downward is, of course, precisely the procedure by which Pascal's triangle is constructed. This isomorphism provides an excellent introduction to the endless fascinating properties of the Pascal triangle.

3



Pascal's triangle gives at once the coefficients for the expansion of binomials—that is, raising $(a + b)$ to any power—as well as the solutions to many problems in elementary probability theory. Note that in Figure 3 the number of shortest paths from the top of the triangle to the bottom row of cells is 1 on the outside border cells, and the numbers increase as you move toward the center. Perhaps you have seen one of those devices based on Pascal's triangle in which a board is tipped and hundreds of little balls roll past pegs to enter columns at the bottom. The balls arrange themselves in a bell-shaped binomial distribution curve precisely because the number of shortest paths to each slot are the coefficients of a binomial expansion.

Susan's algorithm obviously works just as well on three-dimensional grids with cells that are rectangular parallelepids. Imagine a cube that is 3 units on the side, and divided into 27 unit cubes. Consider this a chessboard with a rook in one of the corner cells. The rook can move parallel to any of the three coordinates. In how many ways can it take a shortest path to the cell that is opposite it along a space diagonal?