6.30 Write and test the function that "rotates" 90° clockwise a two-dimensional square array of `int`s. For example, it would transform the array

$$\begin{matrix} 11 & 22 & 33 \\ 44 & 55 & 66 \\ 77 & 88 & 99 \end{matrix}$$

into the array

$$\begin{matrix} 77 & 44 & 11 \\ 88 & 55 & 22 \\ 99 & 66 & 33 \end{matrix}$$

6.31 Write and run a program that reads an unspecified number of numbers and then prints them together with their deviations from their mean.

6.32 Write and test the following function:

```
double stdev(double x[], int n);
```

The function returns the *standard deviation* of a data set of n numbers $x_0, \ldots, x_{n-1}$ defined by the formula

$$s = \sqrt{\frac{\sum_{i=0}^{n-1} (x_i - \bar{x})^2}{n-1}}$$

where $\bar{x}$ is the mean of the data. This formula says: square each deviation `(x[i] - mean)`; sum those squares; divide that square root by `n-1`; take the square root of that sum.

6.33 Extend the program from Problem 6.31 so that it also computes and prints the Z-scores of the input data. The *Z-scores* of the n numbers $x_0, \ldots, x_{n-1}$ are defined by $z_i = (x_i - \bar{x})/s$. They normalize the given data so that they are centered about 0.0 and have standard deviation 1.0. Use the function defined in Problem 6.32.

6.34 In the imaginary "good old days" when a grade of "C" was considered "average," teachers of large classes would often "curve" their grades according to the following distribution:

A: $1.5 \le z$
B: $0.5 \le z < 1.5$
C: $-0.5 \le z < 0.5$
D: $-1.5 \le z < -0.5$
F: $z < -1.5$

If the grades were *normally distributed* (*i.e.*, their density curve is bell-shaped), then this algorithm would produce about 7% A's, 24% B's, 38% C's, 24% D's, and 7% F's. Here the z values are the Z scores described in Problem 6.33. Extend the program from Problem 6.33 so that it prints the "curved" grade for each of the test scores read.

6.35 Write and test a function that creates Pascal's Triangle in the square matrix that is passed to it. For example, if the two-dimensional array `a` and the integer 4 were passed to the function, then it would load the following into `a`:

$$\begin{matrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 1 & 3 & 3 & 1 & 0 \\ 1 & 4 & 6 & 4 & 1 \end{matrix}$$

6.36 In the theory of games and economic behavior, founded by John von Neumann, certain two-person games can be represented by a single two-dimensional array, called the *payoff matrix*. Players can obtain optimal strategies when the payoff matrix has a saddle point. A *saddle point* is an entry in the matrix that is both the minimax and the maximin. The *minimax*