1992 YSU-ACM High School Programming Contest

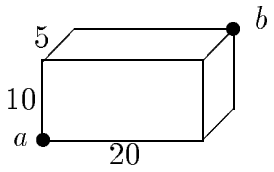
Problem #1

Description: A bug is sitting at vertex a of a rectangular block and wishes to travel to b by walking along the surface. Note that the bug can walk anywhere along the surface, not just along the edges.

Input: The length, width and height of the block each on separate lines.

Output: The length, width, height and shortest distance between a and b.

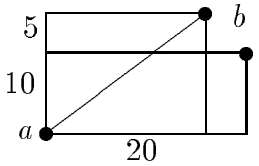
Example:



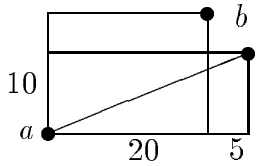
For the block shown here, your output should look like:

```
Length = 5
Width = 20
Height = 10
Minimum distance = 25.0
```

Notes: There are three pairs of possible shortest routes. First unfold the boxes. Then draw lines from a to b and compute the distances. The smaller of the distances is the shortest path. One pair is:



$\sqrt{20^2 + 15^2} = 25.00$



$\sqrt{25^2 + 10^2} = 26.93$

1992 YSU-ACM High School Programming Contest

Problem #2

Description: The fraction 16/64 has the unusual property that if the 6 in the numerator is cancelled with the 6 in the denominator, producing 1/4, the value of the fraction is unchanged. Write a program which displays all fractions less than 1 consisting of two digit numerators and denominators with this property.

Input: None

Output: The fractions with this property in the form nn/dd, one per line.

Notes: Either digit of the numerator can cancel with a like digit in the denominator. Do not cancel $0's$.

1992 YSU-ACM High School Programming Contest

Problem #3

Description: A histogram is a bar graph in which the area of each bar is proportional to the frequency of an item. A histogram can be used to give a visual representation of the letters (A-Z) in a string.

Input: A collection of strings terminated by the string "EOF"

Output: A vertical histogram, representing the occurrences of each letter of the alphabet.

Notes: No lowercase letters will appear in the input file. The characters in the terminating string "EOF" should not be counted. If a letter occurs more than 50 times the histogram should stop at 50. Lines will be no longer than 80 characters. In EBCDIC the letters A-Z are not contiguous. There is space between I-J and R-S.

Example: The following data:

```
THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG
THIS IS AN EXAMPLE OF HOW TO TEST YOUR
HISTOGRAM PROGRAM.  YOU
CAN USE THIS EXAMPLE.
EOF
```

Should produce:

```
                *
    *           *
    *           *
    *           *    *
*   *           *    *
*   *  *        *  ***
*   *  **    * *  ****
*   *  **    * ** ****
*   * ***   ***** ****  **
* ******* ****** **** ***
**************************
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

1992 YSU-ACM High School Programming Contest

Problem #4

Description: Write a program to multiply 2 two-digit numbers as a human would.

Input: Pairs of numbers, one number per line, terminated by both numbers being 0.

Output: The work required to multiply the numbers as a human would. One blank line between each problem.

Example: The following data:
23
15
0
0

Should produce:

```
 23
x15
---
115
230
---
345
```

1992 YSU-ACM High School Programming Contest

Problem #5

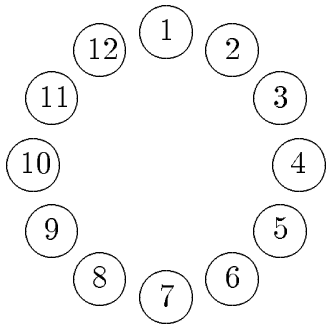
Description: Suppose n people arrange themselves in a circle. Progressing clockwise, starting at 1, around the circle every second person is shot and killed, hence being removed from the circle. The circle of people then close ranks. This is repeated until just two people are left. Assuming you want to remain living, what positions would you want to start at?

Input: The number of people in the circle, n.

Output: The number of people in the original circle on the first line. The order of elimination on the second line. The two positions that you would want to start at on the third line.

Notes: $3 \leq n \leq 50$

Example:



The following data:
12

Should produce:
12
2 4 6 8 10 12 3 7 11 5
9 1

1992 YSU-ACM High School Programming Contest

Problem #6

Description: A system is composed of 3 machines and a repairman responsible for maintaining these machines. Normally, the machines are running and producing a product. At random points in time, the machines fail and are fixed by the repairman. If a second or third machine fails while the repairman is busy fixing the first machine, these machines will wait on the services of the repairman in a first come, first served order. When repair on a machine is complete, the machine will begin running again and producing a product. The repairman will then repair the next machine waiting. When all machines are running, the repairman becomes idle.

Simulate this system for a fixed period of time and calculate the time the machines are busy and the time the repairman is busy. Machines are busy when producing parts and the repairman is busy when repairing machines.

Input: Fifty (50) machine running times in seconds. Fifty (50) machine repair times in seconds. A list of simulation times in seconds terminated by a simulation time of 0.

Output: The simulation number and utilization time of each machine and the repairman.

Notes: If multiple machines break at the same time the machine with the lower number should be taken first. There will be adequate run and repair times to complete each simulation.

Example: The following data: (only 5 run and repair times for this example)

70
25
130
105
30
95
25
55
12
57
100
110
200
0

Should produce:

Case	1	2	3	Repairman	Simulation
1	70	25	100	75	100
2	70	25	110	85	110
3	100	85	130	175	200

Diagram: This diagram is to help explain the simulation process.