UCSD Fall 2006
Programming Contest

Sponsored by

The DiNi Group
Problem 1: Rock, Scissors, Paper

Bart’s sister Lisa has created a new civilization on a two-dimensional grid. At the outset each grid location may be occupied by one of three life forms: Rocks, Scissors, or Papers. Each day, differing life forms occupying horizontally or vertically adjacent grid locations wage war. In each war, Rocks always defeat Scissors, Scissors always defeat Papers, and Papers always defeat Rocks. At the end of the day, the victor expands its territory to include the loser’s grid position. The loser vacates the position.

Your job is to determine the territory occupied by each life form after \( n \) days. The first line of input contains \( t \), the number of test cases. Each test case begins with three integers not greater than 100: \( r \) and \( c \), the number of rows and columns in the grid, and \( n \). The grid is represented by the \( r \) lines that follow, each with \( c \) characters. Each character in the grid is R, S, or P, indicating that it is occupied by Rocks, Scissors, or Papers respectively.

For each test case, print the grid as it appears at the end of the \( n \)th day. Leave an empty line between the output for successive test cases.

Sample Input

```
2
3 3 1
RRR
RRR
RRR
S 4 2
SPSR
SPS
PRSP
```

Output for Sample Input

```
RRR
RRR
RRR
RRRS
RRSP
RSPR
```
Problem 2: Monochrome Inversion

A rectangle with sides parallel to the x-y axes and its left-lower corner at (0, 0) is being painted. The rectangle may be thought of as a flat-screen display with almost infinite resolution; initially the entire rectangle is black. Two numbers are given $0 < h, v < 1$ and then

- A vertical line is drawn dividing the horizontal sides of the rectangle in proportion $h:1-h$ from the left.
- A horizontal line is drawn dividing the vertical sides of the rectangles in proportion $v:1-v$ from the bottom up.
- These two lines divide the rectangle into four smaller rectangles.
- The upper left and the lower right sub-rectangles remain intact.
- The color of the other two rectangles is flipped (from black to white or from white to black) and now each of them is subject to the operation just performed on the bigger rectangle.
- This process continues (in principle) forever.

Given a point in the original rectangle but not on the boundary of any rectangle that turns up in the process of painting, determine the color of the point.

Input contains multiple cases. The first line of each case contains 4 numbers, the length of the rectangle $H$, the height of the rectangle $V$ and then the numbers $h$ and $v$. The next line contains one integer number $n$, the number of points to consider. The following $n$ lines contain two numbers each, the $x$ and the $y$ coordinate of a point.

Organize your output as shown in the sample. For each point from input print the color of the point.

<table>
<thead>
<tr>
<th>Sample input</th>
<th>Output for Sample input</th>
</tr>
</thead>
<tbody>
<tr>
<td>81 32 0.333333333333 0.5</td>
<td>Case 1:</td>
</tr>
<tr>
<td>6</td>
<td>black</td>
</tr>
<tr>
<td>16 30</td>
<td>black</td>
</tr>
<tr>
<td>16 25</td>
<td>white</td>
</tr>
<tr>
<td>16 12.0001</td>
<td>black</td>
</tr>
<tr>
<td>16 11.9999</td>
<td>white</td>
</tr>
<tr>
<td>16 7.987654321</td>
<td>Case 2:</td>
</tr>
<tr>
<td>16 7.0123456789</td>
<td>white</td>
</tr>
<tr>
<td>10 10 0.123456789 0.987654321</td>
<td>black</td>
</tr>
<tr>
<td>2</td>
<td>0.432 0.9876</td>
</tr>
<tr>
<td>9.432 0.9876</td>
<td>white</td>
</tr>
<tr>
<td>0 0 0 0</td>
<td>black</td>
</tr>
</tbody>
</table>
Problem 3: Semi-prime H-numbers

This problem is based on an exercise of D. Hilbert, who pedagogically suggested that one study the theory of $4n+1$ numbers. Here, we do only a bit of that.

An H-number is a positive number which is one more than a multiple of four: 1, 5, 9, 13, 17, 21,... are the H-numbers. For this problem we pretend that these are the only numbers. The H-numbers are closed under multiplication.

As with regular integers, we partition the H-numbers into units, H-primes, and H-composites. 1 is the only unit. An H-number $h$ is H-prime if it is not the unit, and is the product of two H-numbers in only one way: $1 \times h$. The rest of the numbers are H-composite.

For examples, the first few H-composites are:

$5 \times 5 = 25, 5 \times 9 = 45, 5 \times 13 = 65, 9 \times 9 = 81, 5 \times 17 = 85.$

Your task is to count the number of H-semi-primes. An H-semi-prime is an H-number which is the product of exactly two H-primes. The two H-primes may be equal or different. In the example above, all five numbers are H-semi-primes. $125 = 5 \times 5 \times 5$ is not an H-semi-prime, because it's the product of three H-primes.

Each line of input contains an H-number $\leq 1,000,001$. The last line of input contains 0 and this line should not be processed.

For each inputted H-number $h$, print a line stating $h$ and the number of H-semi-primes between 1 and $h$ inclusive, separated by one space in the format shown in the sample.

Sample input

```
21
85
789
0
```

Output for sample input

```
21 0
85 5
789 62
```
Problem 4: Logic Gamble

Logic Gamble is a logic game played with dice. Each die has six faces representing some subset of the possible symbols K, A, N, C, E, p, q, r, s, t. A Well-formed formula (WFF) is any string of these symbols obeying the following rules:

- p, q, r, s, and t are WFFs
- if \( w \) is a WFF, \( \neg w \) is a WFF
- if \( w \) and \( x \) are WFFs, \( Kwx \), \( Awx \), \( Cwx \), and \( Ewx \) are WFFs.

The meaning of a WFF is defined as follows:

- p, q, r, s, and t are logical variables that may take on the value 0 (false) or 1 (true).
- K, A, N, C, E mean and, or, not, implies, and equals as defined in the truth table below.

<table>
<thead>
<tr>
<th>( w )</th>
<th>( x )</th>
<th>( Kwx )</th>
<th>( Awx )</th>
<th>( \neg w )</th>
<th>( Cwx )</th>
<th>( Ewx )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Given a collection of symbols resulting from throwing a set of dice, determine the longest WFF that can be formed from those symbols.

Input consists of several test cases. Each test case is a single line containing a string containing between 1 and 100 of the characters. A line containing 0 follows the last case. For each test case, output a line containing the longest WFF that can be formed using some subset of the letters in the string. If there are several such WFFs, any one will do. If no WFF can be constructed, output a line containing "no WFF possible" as shown below.

Sample Input

```
qKpNq
KKN
0
```

Possible Output for Sample Input

```
KqNq
no WFF possible
```
Problem 5: Mall Crawl

Boston has two giant shopping malls, each enclosing several city blocks. Sam and Jules like to shop and to walk within the malls but they don't like walking between the malls because such walking does not contribute directly to the task at hand: shopping. They would therefore like to know the minimum crossing distance between the malls.

Each city block is a unit square delimited by streets and avenues. Streets run east-west and avenues run north-south. Both are identified by consecutive integers between 0 and 2000. (Lower numbered avenues are west of higher numbered avenues and lower numbered streets are south of higher numbered streets.) Streets and avenues are narrow so their thickness may be assumed to be 0.

Each mall is a contiguous set of complete city blocks. By contiguous we mean that any two blocks are connected by some sequence of blocks such that consecutive pairs of blocks in the sequence share a side. The malls do not intersect and do not surround any empty blocks; that is, the blocks not in any mall are themselves contiguous.

Standard input contains several test cases. Each test case contains the description of the two malls. Each mall's description consists of an integer \( p \geq 4 \), the perimeter of the mall, followed by one or more lines containing \( p \) pairs \((a,s)\) giving the coordinates of the avenue-street intersections contained in the perimeter, in clockwise order. A line containing 0 follows the last test case.

For each case, output a single integer \( d \) -- the minimum walking distance between the malls, assuming that Sam and Jules always walk along streets and avenues.

Sample Input

```
4
0 0 0 1 1 1 0
6
4 3 4 2 3 2
2 2 2 3
3 3
0
```

Output for Sample Input

```
2
```
Problem 6: CSRABBLE

Jane is practicing for a CSRABBLE tournament. She pulls out a handful of CSRABBLE tiles out of a bag, and tries to form the word with the highest possible score. Each tile contains a letter (used to form the word) and a number (its score value). She can use each tile at most once in her word, and she is not required to use every tile. The word that she forms must appear in her dictionary. Her score is the sum of the values of the tiles used in the word.

Note that in CSRABBLE, different tiles with the same letter may have different score values.

To check her work, Jane would like a program to tell her the maximum score possible for the set of tiles. Your task is to write this program.

Input Specification
The first line contains an integer $1 \leq N \leq 100,000$ indicating the number of words in the dictionary. $N$ lines follow, with one dictionary word on each line. Each dictionary word consists of only lowercase letters. The following line contains an integer $1 \leq M \leq 1000$ indicating the number of CSRABBLE hands Jane wants to play. $M$ hands follow. Each hand begins with a line containing an integer $1 \leq P \leq 10$ indicating the number of tiles in the hand. This is followed by $P$ lines, one for each tile. Each of these lines consists of a lowercase letter (the letter on the tile), a space, and an integer $0 \leq V \leq 10$ (the value of the tile).

Output Specification
For each hand in the input, output a line containing the maximum score possible with that hand.

Sample Input
```
2
abcd
hgfe
1
10
a 1
b 2
c 3
d 4
e 5
f 6
g 7
h 8
i 9
j 10
```

Sample Output
```
26
```