

# A+ February 3 - 2018

## Computer Science Competition Hands-On Programming Set

### I. General Notes

1. Do the problems in any order you like. They do not have to be done in order from 1 to 12.
2. All problems have a value of 60 points.
3. There is no extraneous input. All input is exactly as specified in the problem. Unless specified by the problem, integer inputs will not have leading zeros. Unless otherwise specified, your program should read to the end of file.
4. Your program should not print extraneous output. Follow the form exactly as given in the problem.
5. A penalty of 5 points will be assessed each time that an incorrect solution is submitted. This penalty will only be assessed if a solution is ultimately judged as correct.

### II. Point Values and Names of Problems

| Number     | Name       |
|------------|------------|
| Problem 1  | Pottery    |
| Problem 2  | Ruodamudi  |
| Problem 3  | Reverse    |
| Problem 4  | Square     |
| Problem 5  | Quadratics |
| Problem 6  | Holes      |
| Problem 7  | Ornaments  |
| Problem 8  | Family     |
| Problem 9  | Checkpoint |
| Problem 10 | Bags       |
| Problem 11 | Bomb       |
| Problem 12 | Draw       |

# 1. Pottery

**Program Name:** pottery.java

**Input File:** None

3D printing is becoming more and more prominent in today's society. Unfortunately, you don't currently have access to a 3D printer. Instead, you can just print them out with your knowledge of coding! Print out a clay pot.

## Input

There is no input for this problem.

## Output

Output the pot exactly as shown in the example output. There are no blank lines before or after the output, and there are no trailing spaces on any of the lines.

## Example Input File

There is no input for this problem.

## Example Output to Screen

```
  /-----\
 /         \
/           \
|           |
|           |
|           |
|           |
|-----|
| \- / \- / |
|-----|
|           |
|           |
|           |
 \         /
  \-----/
```

## 2. Ruodamudi

**Program Name:** ruodamudi.java

**Input File:** ruodamudi.dat

Since Ruocheng behaved well last year, at New Year's Eve he was visited by Dandamudi who brought an enormous bag of gifts with him! The bag contains  $n$  sweet candies from *the good ol' bakery*, each labeled from 1 to  $n$  corresponding to its tastiness. No two candies in the bag have the same tastiness. The choice of candies has a direct effect on Ruocheng's happiness. One can assume that he should take the tastiest ones — but no, the holiday magic turns things upside down. It is the xor-sum of tastinesses that matters, not the ordinary sum! A xor-sum of a sequence of integers  $a_1, a_2, \dots, a_m$  is defined as the bitwise XOR of all its elements:  $a_1 \oplus a_2 \oplus \dots \oplus a_m$ , here  $\oplus$  denotes the bitwise XOR operation. Dandamudi warned Ruocheng he has more houses to visit, so Ruocheng can take **no more than  $k$**  candies from the bag. Help Ruocheng determine the largest xor-sum (largest xor-sum means maximum happiness!) he can obtain.

### Input

The sole string contains two integers  $n$  and  $k$  ( $1 \leq k \leq n \leq 10^{18}$ )

### Output

Output one number – the largest possible xor-sum.

### Example Input File

4 3

### Example Output to Screen

7

**Note:** A **bitwise XOR** takes two bit patterns of equal length and performs the logical exclusive or operation on each pair of corresponding bits. The result in each position is 1 if only the first bit is 1 *or* only the second bit is 1, but will be 0 if both are 0 or both are 1. In this we perform the comparison of two bits, being 1 if the two bits are different, and 0 if they are the same.

### 3. Reverse

**Program Name:** reverse.java

**Input File:** reverse.dat

In the String class, there exists a function called `substring`. Your task is to do the opposite of the `substring` function. Rather than returning a specified substring within the String, you will output the String with the substring taken out.

#### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow.

Each data set will be one line with a string and two integers, separated by spaces. The first int is the start index of the substring to be taken out, and the second int is the end index.

#### Output

Output the given string, with the substring taken out specified by the given integers. The output will be  $n$  lines, with no leading or trailing white space.

#### Example Input File

```
3
COMPUTER 1 3
SCIENCE 3 7
RULES 3 4
```

#### Example Output to Screen

```
CPUTER
SCI
RULS
```

## 4. Square

**Program Name:** square.java

**Input File:** square.dat

Your mother has asked you to create a template for the squares of a quilt she is making. Her quilt will be based on words! Print out a square of the given word, in the style shown in the example output.

### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will consist of a single line with one word of length  $k$  ( $0 < k < 100$ ).

### Output

Output the square in the format shown in the example output. There are no spaces between data sets. The final square should be both as wide and as tall as the length of the word.

### Example Input File

```
3
one
three
fifteen
```

### Example Output to Screen

```
one
n n
eno
three
h e
r r
e h
eerht
fifteen
i e
f e
t t
e f
e i
neetfif
```

## 5. Quadratics

**Program Name:** quadratics.java

**Input File:** quadratics.dat

After hearing that the quadratic formula is used constantly in calculus, you decide that it would be more efficient to just write a program for it! Write a program that solves the equation shown below:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

The  $\pm$  sign represents two different equations, one with + and one with -. Your answer will be both answers, separated by a comma.

### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will contain three decimals on a single line, representing  $a$ ,  $b$ , and  $c$  respectively.

### Output

Output the two answers to the equation, rounded to the thousandths digit and separated by a comma. The + sign will be the first answer, then the - sign will be the second. The solutions will always be real numbers.

### Example Input File

```
2
1 6 3
4 10 6.1
```

### Example Output to Screen

```
-0.551, -5.449
-1.056, -1.444
```

## 6. Holes

**Program Name:** holes.java

**Input File:** holes.dat

After receiving an impenetrable box, you wonder what's inside! Write a program that finds out how many different disconnected sections the box has, and the total area of the space within the box.

### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will start with two integers  $r$  and  $c$  representing the number of rows and columns of the box, respectively. The next  $r$  lines will represent the box, with  $\#$  representing walls and  $.$  representing spaces. The box will be surrounded entirely by walls.

### Output

Output the number of discrete (disconnected) sections and the total number of spaces in the box, in the format shown in the example output.

### Example Input File

```
2
4 8
#####
#...#..#
#.#.#..#
#####
3 3
###
#.#
###
```

### Example Output to Screen

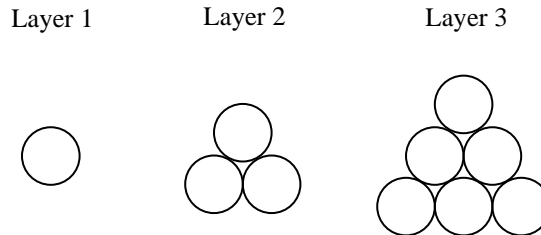
```
2 sections, 9 spaces
1 section, 1 space
```

## 7. Ornaments

**Program Name:** ornaments.java

**Input File:** ornaments.dat

As a worker at the local mall, you have to set out the Christmas ornaments display. The display setup is a triangular pyramid, with the top layer containing one ball, the second from the top containing three, and so on. Each layer's side length will be equal to their layer number. Write a program to determine the total number of ornaments in a pyramid, given the number of layers.



### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will consist of a single line containing one integer, denoting the number of layers in the pyramid.

### Output

Output the total number of ornaments in the pyramid.

### Example Input File

```
2
1
4
```

### Example Output to Screen

```
1
20
```



## 8. Family

**Program Name:** family.java

**Input File:** family.dat

You have been given bits and pieces of your family tree. Your task is to determine if two people are related based on several connections.

### Input

The first line will contain a single integer  $n$  that indicates the number of connections. The next  $n$  lines will consist of a name, a connection, and another name. The connections will be either mother, father, brother, sister, daughter, or son. The next line will contain a single integer  $m$  that indicates the number of test cases. The next  $m$  lines will consist of two names. Your program should determine if the two names are related.

### Output

Output either Related or Not Related, depending on whether they are connected or not. There will be  $m$  lines of output.

### Example Input File

```
3
John brother Susan
Kim mom John
Dave son Jim
2
Jim John
Kim Susan
```

### Example Output to Screen

```
Not Related
Related
```

## 9. Checkpoint

**Program Name:** checkpoint.java

**Input File:** checkpoint.dat

A drone is being tested by finding the shortest path through a maze, reaching every checkpoint on the way in order. Your task is to write a program that finds the shortest path from the start, through every checkpoint in order, and then to the exit, then prints the length of that path.

### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will start with three integers  $r$ ,  $c$ , and  $d$  representing the number of rows and columns of the maze and the number of checkpoints, respectively. The next  $r$  lines will make up the maze, with  $S$  being the starting point,  $E$  being the end point, the numbers 1-9 being checkpoints,  $\#$  being a wall, and  $.$  being an open space.  $S$  and  $E$  also count as open spaces.

### Output

The output will be the length of the shortest path from the start, through every checkpoint in order, and to the exit. There will be  $n$  lines of output with no trailing whitespace.

### Example Input File

```
2
5 8 2
S....1..
.#####.
..2.#...
.#####.
.....E.
1 11 2
S...1...E.2
```

### Example Output to Screen

```
24
12
```

## 10. Bags

**Program Name:** bags.java

**Input File:** bags.dat

When creating a gift bag, you are trying to create a bag that weighs as much as possible with as few items as possible. This is to make it feel like the guests are receiving a lot without having to use as many items in each gift bag. Instead of finding how much to put in each gift bag, write a program that finds the fewest number of items you can put into a gift bag to reach the recommended value.

### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will be three lines, and will start with a single integer  $x$  denoting the number of items. The next line will contain  $x$  integers, indicating the weight of each item. The next line will be a single integer indicating the total weight you are trying to reach.

### Output

Output the smallest number of items that will add up to the weight to be returned. If it is not possible to add up exactly to the weight to be returned, print `Not possible`.

### Example Input File

```
3
10
1 3 3 3 5 7 7 5 5 10
39
10
1 2 3 4 5 6 7 8 9 10
27
1
100
50
```

### Example Output to Screen

```
6
3
Not possible
```

## 11.Bomb

**Program Name:** bomb.java

**Input File:** bomb.dat

You decide to create a game involving a 3d maze with destructible walls, where all the character has to work with is bombs. In order to determine the number of bombs to provide for each level, you need to know the minimum amount necessary to reach the exit and base it off of that. Your task is to write a program that will find the smallest number of bombs necessary to reach the exit. Each bomb can destroy one wall, leaving a blank space in its place.

### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will start with three integers  $f$ ,  $r$ , and  $c$ , representing the number of layers, rows, and columns, respectively. The next  $f$  sets of  $r$  lines will be the maze, with every set of  $r$  lines being one layer of the maze.

### Output

Output the smallest number of bombs necessary to escape the maze. There will be no trailing white space.

### Example Input File

```
2
2 3 3
S##
##E
###
#.#
#..
###
1 2 10
S#.####.#E
..##..###.
```

### Example Output to Screen

```
1
5
```

## 12. Draw

**Program Name:** draw.java

**Input File:** draw.dat

You're bored during class one day. Instead of drawing shapes all over your paper, you decide to write a program that will do it for you! Write a program that, when given the type of shape and the dimensions, will draw the specified shape. The program will also be able to either leave the shape empty or fill it in. The shape names and examples are shown below:

|                |                |                             |                               |
|----------------|----------------|-----------------------------|-------------------------------|
| rectangle      | ##<br>##       | ###<br># #<br>###           | ###<br>###<br>###             |
| left triangle  | #<br>##<br>### | #<br>##<br># #<br>####      | #<br>##<br>###<br>####        |
| right triangle | #<br>##<br>### | #<br>##<br># #<br>####      | #<br>##<br>###<br>####        |
| diamond        | #<br># #<br>#  | #<br># #<br># #<br># #<br># | #<br>###<br>#####<br>###<br># |

The rectangle can be any number of rows and columns. The left triangle, the right triangle, and the diamond will always have the same number of rows and columns. For the diamond, the number of rows and columns will always be odd.

### Input

The first line will contain a single integer  $n$  that indicates the number of data sets that follow. Each data set will be one line that starts with the shape name. If the shape is a rectangle, the name will be followed by two integers,  $r$  and  $c$ , representing the number of rows and columns respectively. If not, then the shape name will be followed by one integer, denoting the number of both rows and columns. The line will end with either  $y$  or  $n$ ,  $y$  meaning that the shape is filled in and  $n$  meaning that the shape is empty.

### Output

You will print the specified shape of the specified size, either shaded or unshaded as denoted by the letter at the end of the line. There are examples of the shapes in the table above. There are no lines of whitespace between data sets.

### Example Input File

```
3
rectangle 3 5 n
right triangle 4 n
diamond 7 y
```

### Example Output to Screen

```
#####  
#      #  
#####  
      #  
      ##  
      # #  
#####  
      #  
      ###  
#####  
#####  
#####  
#####  
      ##  
      #
```