# ND Programming Contest, Fall 2009

ND ACM Computer Club

Saturday October 10, 2009

# #1: The Hardest Problem Ever

## The Problem

Julius Caesar lived in a time of danger and intrigue. The hardest situation Caesar ever faced was keeping himself alive. In order for him to survive, he decided to create one of the first ciphers. This cipher was so incredibly sound, that no one could figure it out without knowing how it worked.
You are a sub captain of Caesar's army. It is your job to decipher the messages sent by Caesar and provide to your general. The code is simple. For each letter in a plaintext message, you shift it five places to the right to create the secure message (i.e., if the letter is 'A', the cipher text would be 'F'). Since you are creating plain text out of Caesar's messages, you will do the opposite:
Cipher text: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Plain text: V W X Y Z A B C D E F G H I J K L M N O P Q R S T U Only letters are shifted in this cipher. Any non-alphabetical character should remain the same, and all alphabetical characters will be upper case. [1]

## Sample Input

Input to this problem will consist of a (non-empty) series of up to 100 data sets. Each data set will be formatted according to the following description, and there will be no blank lines separating data sets. All characters will be uppercase. A single data set has 3 components: //1.Start line - A single line, "START" //2.Cipher message - A single line containing from one to two hundred characters, inclusive, comprising a single message from Caesar. //3.End line - A single line, "END" Following the final data set will be a single line, "ENDOFINPUT".

```
START
NS BFW, JAJSYX TK NRUTYFSHJ FWJ YMJ WJXZQT TK YWNANFQ HFZXJX
END
START
N BTZQI WFYMJW GJ KNWXY NS F QNYYQJ NGJWNFS ANQQFLJ YMFS XJHTSI NS WTRJ
END
START
IFSLJW PSTBX KZQQ BJQQ YMFY HFJXFW NX RTWJ IFSLJWTZX YMFS MJ
END
ENDOFINPUT
```

## Sample Output

For each data set, there will be exactly one line of output. This is the original message by Caesar:

[1] Note: This problem was taken from the 2002 South Central USA Regional Programming Contest

I WOULD RATHER BE FIRST IN A LITTLE IBERIAN VILLAGE THAN SECOND IN ROME
DANGER KNOWS FULL WELL THAT CAESAR IS MORE DANGEROUS THAN HE

# #2: Going Green

## The Problem

In an effort to try to understand energy consumption in their dorm some students decide to simulate how often lights are used. Lets say there are n students conducting the experiment using their n dorm rooms(numbered 1 through n). The system they devised for testing is as follows. First the light in each room is turned off. At some point during the day each of the students will walk down the hallway. The student who lives in room i will first flip the switch in their room and then travel i more rooms to flip another switch, repeating this process until they reach the end of the hallway. Your job is to figure out how many lights will be on at the end of the day.

For example if there are 20 students, the student who lives in room 5 will flip the switches in rooms 5, 10, 15, and 20.

Here is an explanation of the second sample test case when there are 5 students. The switch in room 1 is only flipped by the person who lives there so it is on. The switches in rooms 2 3 and 5 are flipped by the person who lives in each and the person from room 1, so they are all off. The switch in room 4 is flipped by the people from rooms 1, 2, and 4, so it is on at the end of the day. Rooms 1 and 4 have their lights on at the end of the day, so the proper output is 2.

## Sample Input

The input will begin with the number of test cases. For each test case there will be a single number representing how many students are participating in the experiment.

```
3
1
5
20
```

## Sample Output

The output will consist of one number saying how many lights are on at the end of the day for each test case.

```
1
2
4
```

# #3: Squirrel Hunting

## The Problem

Due to an abundance of acorns on campus in the last year, we are facing a major infestation of squirrels across campus. The administration has divided up campus into an n by n grid and has published reported squirrel populations in each sector. You have been issued a golf cart and net to help reduce the squirrel population. You start out in the northwest corner of campus and must return the golf cart to the southeast corner of campus. In order to ensure you don't try to steal the golf cart it is set to self-destruct if you go any direction other than east or south or try to leave the grid.

In order to maximize the number of squirrels you can catch, you must find the route which maximizes the number of squirrels you will see on the way.

## Sample Input

The input will begin with an integer n representing the size of the grid. The rest of the input will consist of a list of all the squirrel populations in each sector.

```
5
1 5 2 3 6
4 3 2 1 2
3 8 4 2 1
0 5 2 3 4
3 1 4 2 1
```

## Sample Output

The output will be a single number which is the sum of the squirrel populations in the sectors visited by the optimal route.

```
32
```

| | | | | |
|---|---|---|---|---|
| Start 1 | 5 | 2 | 3 | 6 |
| 4 | 3 | 2 | 1 | 2 |
| 3 | 8 | 4 | 2 | 1 |
| 0 | 5 | 2 | 3 | 4 |
| 3 | 1 | 4 | 2 | 1 Finish |

Figure 1: Sample Input Squirrel Population Grid with optimal path highlighted

# #4: Coaching Football

## The Problem

While Charlie Weis has had some success with the Notre Dame football team this year, he is still looking for ways to improve. He is not satisfied with barely winning games in overtime. During the most recent football game against the Washington Huskies, Charlie noticed that his team was racking up the points in field goals but failing to make touchdowns. Charlie hopes to fix this for future games.

Your task is to write a program that gives Coach Weis all possible combinations of how the football team could have earned a final score. Note that we are not considering points earned for a safety, since they happen so rarely in games. Just in case you've been spacing out during the football games, here are the different ways you can score points in a football game:

Field Goal = 3 points
Touchdown = 6 points
After a touchdown:
Extra point = 1 point
Two-point Conversion = 2 points

## Sample Input

Input will consist of a series of lines, each representing a final score. The last integer will be "0", which indicates the end of the list, and your program should quit upon receiving this.

```
7
3
10
9
14
0
```

## Sample Output

Output should consist of a series of lines. Each line will contain the number of possible combination of events that will lead to the final score.

```
1
1
1
2
3
```

# #5: Sprinkler Dodging

## The Problem

While trying to get to class one morning you realize that all of the sprinklers on campus are turned on. You are carrying a paper that you have to turn in when you get to class so you can't afford to get wet. Your job is to try to find a path from your dorm to your class that stays on campus and doesn't come within range of the sprinklers.

Campus will be represented by a 20 by 20 box, with your dorm located at (0,0) and your classroom at (20,20).

## Sample Input

Input will begin with the number of test cases. Each test case will start with the number of sprinklers. Each sprinkler will be represented by 3 integers the x coordinate, the y coordinate, and the radius it shoots water.

```
3
1
19 19 1
5
20 0 5
15 5 5
10 10 3
5 15 5
0 15 5
2
10 10 5
5 15 5
```

## Sample Output

For each test case, your program should output whether or not there is a path from your dorm to class without getting wet.

```
no
yes
yes
```

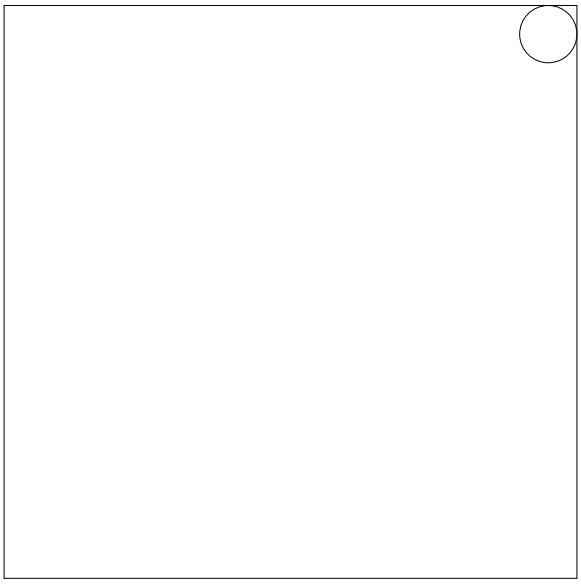

Figure 2: Sample input 1 sprinkler system

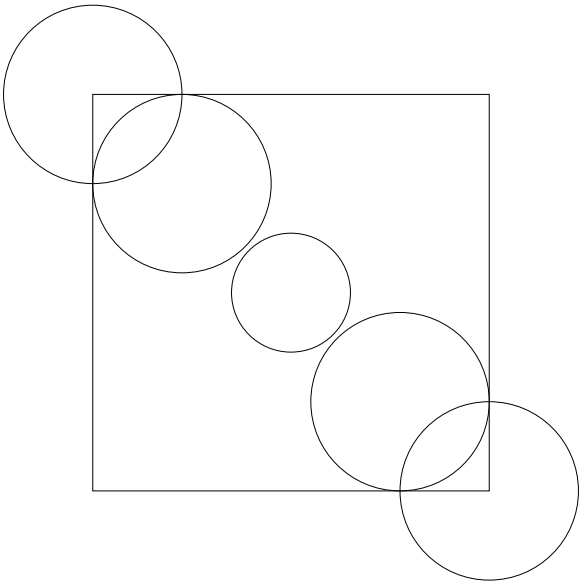

Figure 3: Sample input 2 sprinkler system

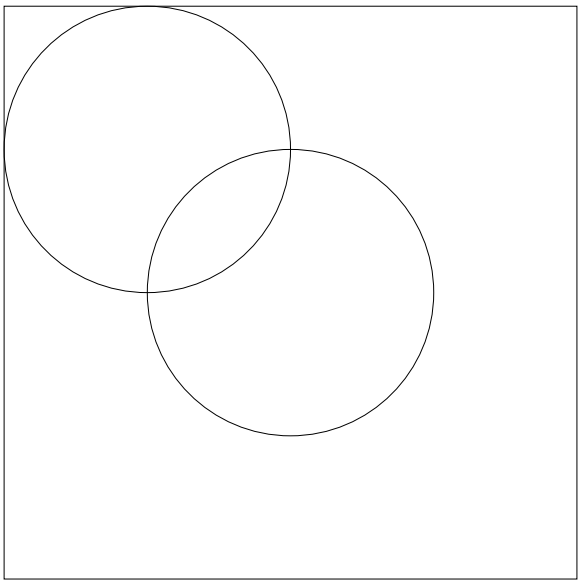

Figure 4: Sample input 3 sprinkler system