COMPUTER SCIENCES

George Fox University
H.S. Programming Contest
Division III
2025

General Notes

1. Do the problems in any order you like. They do not have to be done in order
(hint: the easiest problem may not be the first problem… but probably is)

2. Scoring: The team who solves the most problems in the least amount of time with the least submissions wins. Each wrong submission will receive a 20 min time penalty that will only be added to the time score once the problem has been successfully solved. Time is calculated for each problem as the total time from the start of the contest to the time it was solved.

3. There is no extraneous input. All input is exactly as specified in the problem. Integer inputs will not have leading zeros.

4. Your program should not print extraneous output. Do not welcome the user. Do not prompt for input. Follow the form exactly as given in the problem.
(hint: spaces? No spaces? What does spec say!)

5. All solutions must be a single source code file. *(no spaces in filenames)*

This page intentionally left blank

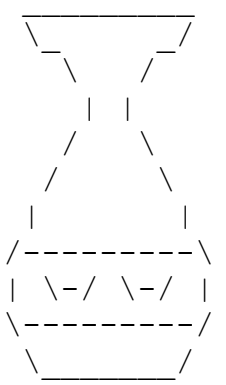
1. Pottery

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
none

Output
Output the pot exactly as shown in the example output. *(hint: the image is not rectangular, there are no trailing whitespaces at the end of each line)*

Example Input
```
none
```

Example Output to Screen


This page intentionally left blank

2. Hopper

Of all the rivalries in the animal kingdom, that of the grasshopper and the frog gains infamy as the most pervasive and devastating. Each year, the lives of grasshopper families are disrupted as large frogs hop in to eat them whole, never to be seen again.

You and your isopod friends have been contracted to assassinate The Hopper, a large bullfrog weighing well over 20 pounds. His reign of terror has continued unabated for several years, and it's time for you to end it.

Hearing of your plot, he goes into hiding and shuts off his social media. You and your crew of isopod assassins must design a computer program that takes in everything that The Hopper has ever sent online in order to determine whether the leads you have received about his location are accurate.

Input

The first input will contain a single integer `n` that represents the number of data sets that follow. Each data set will contain a single word that is the name of a location that The Hopper has been reported to have been spotted at, followed by a caption or message from one of his social media accounts.

Output

For each data set, output either "`Good Lead`" if the lead location was mentioned in the text, or "`Bad Lead`" if the lead location was not mentioned. At the end of the text, output the total number of good leads that you had.

Example Input

```
5
Barn Wassup party people I'm heading to the barn
Bucket Hey man I need some cash rn
cornhole Brother, I am moving over to with the frogs at cornhole
Grass Loving the grass out here today! Gotta be my fav spot
street See you out in the street my guy.
```

Example Output to Screen

```
Good Lead
Bad Lead
Good Lead
Good Lead
Good Lead
4
```

This page intentionally left blank

3. Seeds

After your isopod gang makes The Hopper "disappear", you acquire his large quantity of seeds which he kept in storage. For insects, the main medium of exchange is seeds, so he had a large pull with all of the insects that he decided to do business with instead of eating.

You are having trouble putting together a plan to get them all back to your home base, so you all decide to hire out a group of ants to move them all into storage for you. Their preferred form of compensation is a small cut of the seeds that you need transported.

The rate they charge is 100 seeds per pound per inch travelled. In other words, for each pound of seeds carried one inch they require 100 seeds as payment.

Find the number of seeds that you need to pay the ants as compensation based on distance and time.

There are 4000 seeds per pound.

Input

The first input will contain a single integer `n` that represents the number of data sets that follow. Each data set will contain the number seeds in a seed pile, and the distance in inches from the pile to the storage location.

Output

Output the total amount of seeds paid as compensation for the total pounds per inch traveled: "`n seeds paid in compensation for x lbs/inch total`". Round seeds up to the next whole number, and round lbs/inch to two decimal places.

Example Input

```
5
12038 23
1237123 12
123870 232
1238 12
55555 105
```

Example Output to Screen

```
6922 seeds paid in compensation for 6921.85 lbs/inch total
371137 seeds paid in compensation for 371136.90 lbs/inch total
718446 seeds paid in compensation for 718446.00 lbs/inch total
372 seeds paid in compensation for 371.40 lbs/inch total
145832 seeds paid in compensation for 145831.88 lbs/inch total
```

This page intentionally left blank

4. Kilometer

After parlaying with your isopod deity, you determine that there is a near infinite source of food for your colony nearly a kilometer away. You decide to take the next few weeks to commission a group of your friends to draw up a set of maps and potential routes to get to the new food source of sustenance and await their findings.

After much waiting you finally assemble the group to review their work. They present to you various paths to reach your end destination, each one slightly different than the other.

You have no idea how to pick the right one, so you decide to choose the route that takes you to the site with the largest food pile. The end destinations of each path are shown as a 5x5 square of characters printed on paper.

Input

The first input will contain a single integer `n` that indicates the number of character grids that will follow. Each character grid will be followed by a blank line. Each piece of food in a clump will be represented by the character `"&"`. Food pieces will be part of the clumps that they are directly adjacent to, not at their edge. There will always be at least one piece of food.

Output

Output the number of food pieces in the largest clump of food that is found for each end destination.

Example Input

```
3
#&&##
##&##
&####
#&&##
###&#

#####
#####
##&##
#####
#####

####&
#&###
&&#&#
#&#&#
#####
```

Example Output to Screen

```
3
1
4
```

This page intentionally left blank

5. Caffeine Anhydrous

The greatest of discoveries, Caffeine!

As you gather your tribe and arrive in the promised land you discover that the food promised to you was no ordinary food, but a large supply of sugar, salt, and caffeine! Though in amounts large enough both salt and caffeine will kill you, the combination of small parts of the two in conjunction with large amounts of sugar has given you enough energy to kick-start the growth of a new civilization.

Input

The first input will contain a single integer `n` that indicates the number of data sets that follow. Each data set will contain three integers, which correspond in order to the sugar, salt, and caffeine content of a portion of the food.

Output

For each data set, output a matrix composed of the word `isopod` as its border and the character `&` as its interior with a side length equal to the minimum of the three integer values. Each outputted matrix will be followed by a blank line.

Example Input

```
3
1 2 3
3 4 5
6 7 9
```

Example Output to Screen

```
i

iso
s&p
ido

isopod
s&&&i
i&&&s
d&&&o
o&&&p
posido
```

This page intentionally left blank

6. Quick Trip

Hundreds of years later the universe is at your fingertips. The secrets to eternal life have been revealed to you, and all mysteries are at hand. You have done what was one thought impossible. Isopods rule all.

Given your omniscient knowledge of every inch of the universe, find the shortest path to the center of the universe.

Input

The first input will contain a single integer `n` that indicates the number of data sets that follow. Each data set will contain the integers `x` and `y` which represent the width and height of the matrix to follow. The matrix represents a rough outline of the universe. The integer value of `1` represents an available path to travel. The center of the universe will always be the bottom right corner of the matrix and your position is always the upper left corner of the matrix. If there is no possible path return `-1`.

Output

For each data set, output the distance units travelled if you were to take the shortest path to the center of the universe.

Example Input

```
2
5 5
11100
10100
10100
11111
10001
2 6
10
10
11
01
01
01
```

Example Output to Screen

```
8
6
```

This page intentionally left blank

7. Screenplay

You're working for a film studio to make some money to fund your independent film making and they've given you tons of monotonous busy work to do. This is fine usually as you can pretend to work while actually working on your screenplay. The issue is today you have been assigned the labor intensive job of sorting all the movies they have on dvd in alphabetical (ignoring case) order. You need to find out the least number of swaps you can make to sort the list of movies given.

Input

The first input will contain a single integer `n` that represents the number of data sets that follow. Each data set will contain a list of strings, each of which will denote the name of one of the movies that you need to sort (alphabetically by title, ignoring case), all separated by spaces. There will not be any duplicate names of movies within a data set.

Output

For each data set output the smallest number of swaps needed to sort the list of movies alphabetically.

Example Input

```
3
TheDarkKnight BatmanBegins TheDarkKnightRises Tenet Interstellar ThePrestige
DjangoUnchained KillBillVol1 KillBillVol2 TheHatefulEight ReservoirDogs
CaptainMarvel Thor Avengers
```

Example Output to Screen

```
4
1
2
```

This page intentionally left blank

8. Casting

You have finally begun the laborious process of casting for your movie. You have a list of actors with their years of experience and how much you'd have to pay them. You need to determine what is the highest combined years of experience you can get for a given maximum amount of money you can pay out (in thousands).

Input

The first input will contain a single integer `n` that represents the number of data sets that follow. Each data set will start with 2 integers, `m` and `a`, denoting the number of casting options and the maximum amount (in thousands) you can pay out to the cast of this movie. The next line will contain `m` integers, denoting the years of experience for each possible cast member. The following line will contain `m` integers, denoting the amount of money you'll need to pay each cast member for the movie.

Output

For each data set, output the maximum combined years of experience you can afford to get for the cast.

Example Input

```
2
6 100
5 3 8 2 4 2
70 45 75 25 50 30
5 80
1 1 2 2 3
30 35 40 45 50
```

Example Output to Screen

```
10
4
```

This page intentionally left blank

9. Premiere

Your movie, “New Reality” premiers tomorrow! Of course all the cast (now celebrities because of your movie) and family members will be attending in all their crazy clothes, and you need to coordinate the photographers and red carpets so that there will be enough for everyone to take plenty of pictures. Determine the minimum number of photographers so that no photographer will have to take photos of more than one celebrity at once. If one celeb leaves the photo area at the exact time another arrives, then one photographer will be enough. All celebs will arrive between 6:00 pm and 12:00 am, and this will only be one night.

Input

The first input will be a single integer `n` that represents the number of data sets that follow. Each data set will begin with an integer `m`, denoting the number of celebs in this data set. On the next line, each celeb entry will consist of a time stamp, the arrival time of that celeb, and an integer, denoting the number of minutes that that celeb will spend being photographed, with a space between these two values, and a space between each entry.

Output

For each data set, output the integer denoting the minimum number of photographers needed to take pictures at your premiere, so that each photographer is only taking pictures of one celeb at a time.

Example Input

```
2
5
7:00 10 7:05 7 7:15 15 7:13 10 7:30 10
7
8:00 13 8:15 15 8:05 17 8:22 4 8:09 3 8:31 12 8:26 6
```

Example Output to Screen

```
2
3
```

This page intentionally left blank

10. Star Student

Your teacher needs help finding the student with the highest grade in her class. Being the only computer science student in the class, she asks you for help to make a program. Given all her students' grades, write a program to output the name of the student with the highest grade.

Input

The first input will be a single integer `n` that indicates the number of data sets that follow. Each data set will start with a single integer `x` denoting how many students are in that data set. The next `x` lines will contain a string and an integer that represent the student's name and grade.

Output

For each data set output the name of the student with the highest grade. There will only be **ONE** student with the highest grade. Grades will be in the range of (0-100).

Example Input

```
2
3
Timmy 94
Jake 82
Luke 13
4
Alex 99
Duke 84
Nathan 100
William 78
```

Example Output to Screen

```
Timmy
Nathan
```

This page intentionally left blank

11. Coins

You're a wealthy human, and wealthy humans don't count their own pocket change. Wealthy humans have their robot butlers count their pocket change. However, you are slightly frugal and bought a robot butler that still needs to be programmed. Luckily, you are a computer scientist. You job is to write a program for your robot butler. The program will determine if you have a certain amount of change given the coins in your pocket.

Input

The first input will be a single integer `n` that indicates the number of data sets that follow. Each data set will consist of 2 lines of data. The first line of the data set has two integers. The first integer represents the desired value to be made in cents, and the second integer, `c`, represents the number of coin denominations there are. On the following line, there will be `c` integers representing the value, in cents, of the coins available. All coins can be used an infinite number of times.

Output

For each data set, output "`Possible`" or "`Not Possible`" based off whether the desired value can be made with the available coins.

Example Input

```
2
101 2
5 50
10 3
2 3 4
```

Example Output to Screen

```
Not Possible
Possible
```

This page intentionally left blank

12. Joe's Day

Joe's schedule today is totally booked. On days like this, he likes to know exactly what time he'll be finished. Assuming that Joe will either be working or driving nonstop throughout his workday, use his schedule to predict when he will finally arrive home from his workday.

Input

The first input will contain a single integer `n` that indicates the number of data sets that follow. Each data set will start with an integer `s` representing his car's speed in mph, then an integer `t` representing the number of tasks in his day to get home followed by a time in the format HH:MM AM/PM, representing the time Joe begins the first item on his to do list. The following `t` lines will each contain one item of Joe's to do list, formatted by the items name, followed by either a distance in miles or a time in minutes, separated from the task name by a comma. The time it takes to complete that item will either be the time it takes for Joe's car to travel the provided distance, or the time listed.

Output

For each data set, output the time joe will arrive home to the nearest minute in the following format : `"Joe will arrive home at HH:MM AM/PM"`.

Example Input

```
2
10 5 07:00 AM
GO TO JOB 1, 20 MILES
DO JOB 1, 30 MINUTES
GO TO JOB 2, 40 MILES
DO JOB 2, 45 MINUTES
GO HOME, 10 MILES
50 3 05:00 PM
GO TO STORE, 25 MILES
SHOP, 15 MINUTES
GO HOME, 25 MILES
```

Example Output to Screen

```
Joe will arrive home at 03:15 PM
Joe will arrive home at 06:15 PM
```

This page intentionally left blank