

Stacking Waffles

Author: Ryan Elian

File: waffles.cpp / waffles.c / waffles.pas

Memory: 32 MB

Time Limit: 1 sec.

Descriptions:

Y. R. Halim loves to eat. One of his favorite munchies is waffle (fresh, gold and tender ones served with sweet and rich maple syrups; best while hot!). Waffles came in various sizes, but all of them usually are rectangles (Well, that makes the difference of waffles and pancakes to be only the shape! Pancakes are round, by the way).

In an all-you-can-eat promo at a restaurant, Halim can get himself on a plate, a stack of waffle as much as he can. Unfortunately, waffles can only be stacked from the largest to the smallest, since stacking big waffle over equal-sized or smaller waffles can unbalance the stack and make the whole waffles fall down. Therefore Halim must be careful when choosing waffles, because after he put a waffle in his plate, he cannot put it back.

Each waffles defined as $0 < W_i, B_i \leq 100000$, $W_i \times B_i$ as Width x Breadth. Waffles that are bigger have either larger W_i or B_i than the smaller waffles. Determine how many maximum waffles Halim can put into his plate in a single take.

Input Formats:

First line consists of an integer $0 < N \leq 2009$, which denotes the number of waffles in line. The next N lines consist of two integers W_i and B_i .

Sample Input:

```
6
5 5
1 1
2 2
4 10
3 4
2 3
```

Sample Output:

```
4
```

Output Explanation:

A single integer. Halim takes waffles #1, #5, #6 and finally #2. So in total, there are four waffles in the stack.

Pancake Profits

Author: Dwika Putra

File: profits.cpp / profits.c / profits.pas

Memory: 32 MB

Time Limit: 1 sec.

Descriptions:

Mr. D is a retired high-school-programmer. He is now working in Australia for a newly appointed CEO of a pancake producing enterprise, well-known for their best invention, the pancake with special flavor like Carbonara Pancake or Marinara Pancake. The CEO is very obsessed with round things, because he likes Pancake very much. The financial director came and told the CEO about the growth of their profit in the last 30 years, in a day-by-day bases. The enterprise's profit is as follows:

- The first day profit is one full dollar
- The second day profit is two times of the first day
- The third day profit is third of the second day; and so on.

But since the CEO is very obsessed with round things, he only wants the number of zeroes at the end of their profit. The financial director won't bother with counting the zeros, so he appointed Mr. D to make a program for it. Because Mr. D is now a retired high-school-programmer and he thought this is a high-school-programmer level problem, he asked you to make it, while he will be in his armchair, playing NBA Live 09 in his PSP and eating Carbonara Pancake with extra cheese.

Input Formats:

Input will consists of several lines, the first line is $1 \leq X \leq 1000$. Each of the following X lines have single integer $1 \leq Qi \leq 10000$, representing the profit of the day asked by the CEO.

Output Formats:

Output will consists of X lines, each line is a single integer telling the number of zero in the end of the profit for the specific day.

Sample Input:

7
1
3
5
10
8

Sample Output

0
0
1
2
1

Meet The Pancake Lovers

Author: Dwika Putra

File: lovers.cpp / lovers.c / lovers.pas

Memory: 32 MB

Time Limit: 1 sec.

Descriptions:

After being fired for illegal outsourcing from the pancake company, Mr. D is now working as a low-paid, part time farmer in a suburb called Bundoora in Melbourne. One day, the lady of the farm wanted to hold an annual meeting, for she is the Leader of the *Pancake Lovers*, a social group dedicated to pancakes, with slogan "*Life is like cooking a pancake, easier said than done*". She wants to make several 1-on-1 meetings with the candidates of the guest (she doesn't want anyone get embarrassed for low-knowledge about pancakes), but she cannot arrange her meeting time by herself; because she is a real kind Australian that honors senior (older people will be treated first). Every minute, a new people will come, and they will rearrange themselves due to their age. No one is serviced before all candidates come.

And, Mr. D who is in desperate need of money, asked if he could get some extra money by helping her. After being approved to do so, he once more appointed you to do the hard work. Intimidated by a farming fork in his hands, you agreed.

Input Formats:

The first line consists of a single integer $1 \leq N \leq 10000$, which indicates the number of the guests coming. The next N lines will consist of two integers, $1 \leq A \leq 100$ and $1 \leq X \leq 10^9$, which respectively show the age of the person and the registration number given. Last line of the input is an integer $1 \leq Q \leq 10^9$.

Output Formats:

The output consists of exactly N lines; each line represents the position of the candidate with the registration number Q every minute. If in a specific minute, a candidate with number Q does not present: output **-1**. If no candidate with number Q , then print **ERROR** once. Number 1 represents the front of the queue.

Sample Input 1:

5
18 1001
29 1002
7 20156
90 1108
1 20
1002

Sample Output 1:

-1
1
1
2
2

Sample Input 2:

5
18 1001
29 1002
7 20156
90 1108
1 20
1001

Sample Output 2:

1
2
2
3
3

Prison Survival

Author: Dwika Putra and Ryan Elian

File: *prison.cpp / prison.c / prison.pas*

Memory: 32 MB

Time Limit: 1 sec.

Descriptions:

Pancake Lovers were upset with the pancake company that Mr. D had worked with. Therefore, the group launched a massive coup d'état against the company, armed with enormous equipments of farming forks. But the company (with all its profits) deployed air units armed with pancake bombs (think of it as deadlier pineapple grenades), thus the strike failed and the *Pancake Lovers* imprisoned. So, Mr. D is now imprisoned in the V State Prison. (Don't worry, only until ICC is over)

The V State Prison is an extremely cruel prison, where every day, some people are not given dinner. Before dinner, all prisoners were broken into several groups and given tickets with number in it, one for each prisoner. Here are the table rules:

1. The tickets are labeled 2 to $N+1$ (where N is the number of prisoner in that group)
2. If for any prisoner, their number on ticket (NoT) divisible by any other prisoner's ticket, then they got dinner.
3. For those who haven't got dinner after #2, they must compare their NoT to each other. Everybody received half dinner portions except the one with biggest number.

Even though Mr. D is a very bad guy, he is extremely gentle. So he always let smaller people take tickets before him. Since Mr. D is very tall, he always received the last number. Make a program, just to know whether Mr. D got his dinner or not; then laugh-out-loud (read: LOL) if the output is NO. But the V State Prison sentinel asked you to update your program a little bit, to know how many portion of the food is needed for every group.

Input Formats:

The first line will be a single Integer $1 \leq N \leq 10000$. The next N lines each consists of single integer $10 \leq X \leq 10000$

Output Formats:

The output consists of exactly N lines. Each line will consist of a string either **YES** or **NO**. Where **YES** means Mr. D received his dinner and **NO** if otherwise; and a floating-point number denoting food portion amount required for the prisoners that night.

Sample Input:

```
3
10
12
11
```

Sample Output:

```
NO 7.5
NO 8.5
YES 8.0
```

Factories

Author: Ryan Elian

File: fact.cpp / fact.c / fact.pas

Memory: 32 MB

Time Limit: 1 sec.

Description:

Cindy was assigned her first programming problem! The problem is quite simple actually; she had to create a code that list all the factors of a number inputted. But Cindy is a perfectionist. She wanted a much better program, which function is to list all prime factors of the number and show how many of each of that prime factors required in forming the number. Because Cindy just learned programming, she had difficulties in making it! (Even though, she persists on making it.) Please help Cindy!

Input Description:

An integer $0 \leq N \leq 100000$.

Output Description:

A series of line in which each line has two integers: the first one denotes the prime factors of input **N** and the second one denotes the number of that prime factor exists in the number. If no prime factors exist, output 1. Output the numbers descending by prime factors.

Sample Input:

90

Sample Output:

5 1
3 2
2 1

Arranging by Metadata

Author: Ryan Elian

File: meta.cpp / meta.c / meta.pas

Memory: 32 MB

Time Limit: 1 sec.

Description:

Metadata are 'extra' information in PC files and documents. Metadata are very useful and used extensively in libraries (e.g. Music Libraries use ID3 Tag to identify MP3 file authors, date of creation, album, etc.). PC file managers usually can execute actions based on these metadata.

Let there be a folder in a computer having files that have these following metadata:

Metadata	Data Representation
File Name	A Text, up to 200 characters long. Characters, numbers and symbols only.
File Type	A Text, up to 200 characters long. Characters, numbers and symbols only.
Date Modified	<Date>(SPACE)<Month>(SPACE)<Year> All elements are integers, ranging 0...10000: (e.g.) 18 02 2009
File Author	A Text, up to 200 characters long. Characters, numbers and symbols only.
File Size (in Bytes)	An Integer, guaranteed to fit in a signed 32-bit integer.

Assume you are using terminal to see the files. You have these following commands:

Command	Details
SHOW	Followed by a space then either: <ul style="list-style-type: none">- A word ALL, after this command: print all the files metadata, one file per line.- An integer i, after this command: print the i-th file metadata.
SORT	Followed by a space then one of the following word: <ul style="list-style-type: none">- NAME: sort the files ascending , lexicographically by the File Name- TYPE: sort the files ascending, lexicographically by the File Type- DATE: sort the files ascending by Date Modified- AUTHOR: sort the files ascending, lexicographically by File Author- SIZE: sort the files ascending by File Size value If there are files with duplicate metadata to be sorted, preserve orders of the files.

EXIT	End the command prompt, ignore commands below this command.
------	---

Your task is to read the folder content and then perform actions as commanded. Files in the folder originally may be unsorted. (Lexicographically, symbols came before characters.)

Input Description:

An integer $0 < N \leq 100000$, which indicates the number of files in the folder. The next N lines consist of the five metadata in order according to the table above, space separated. The remaining input data is lines of command which formats can be referred to the table given.

Output Description:

Outputs are dependent to the commands given.

Sample Input:

```
4
Winner's_Table PDF 20 3 2009 ICC2009 200000
English_Essay DOC 1 2 2009 Rivaldi 26000
ICCLogo_(Large) TIFF 23 12 2005 Nichole 70000000
GCC_MinGW TAR.GZ 16 2 2009 TDragon.net 46000000
SHOW 1
SORT DATE
SHOW 3
SORT SIZE
SHOW ALL
EXIT
SHOW 1
EXIT
```

Sample Output:

```
Winner's_Table PDF 20 3 2009 ICC2009 200000
GCC_MinGW TAR.GZ 16 2 2009 TDragon.net 46000000
English_Essay DOC 1 2 2009 Rivaldi 26000
Winner's_Table PDF 20 3 2009 ICC2009 200000
GCC_MinGW TAR.GZ 16 2 2009 TDragon.net 46000000
ICCLogo_(Large) TIFF 23 12 2005 Nichole 70000000
```