WOBURN CHALLENGE

2015-16 Online Round 1

Friday, October 16th, 2015

Junior Division Problems

Automated grading is available for these problems at:
wcipeg.com

For more problems from past contests, visit:
woburnchallenge.com

Problem J1: Do You Know Your Woburn Colours?

10 Points / Time Limit: 2.00s / Memory Limit: 16M *Submit online: wcipeg.com/problem/wc151j1*

Welcome to the first Woburn Challenge in over a decade! We hope you find these contests fun and challenging. Let's start by getting to know Woburn a little more. Woburn's school cheer is the legendary *Wildcat Song*, and it starts like this:

```
CHEER! CHEER! FOR WOBURN...
CHEER WITH ALL YOUR MIGHT.
CHEER FOR OUR COLOURS,
RED AND BLUE AND WHITE.
```

A true Woburnite obviously knows Woburn's colours by heart.
Do you think you have what it takes to identify them?

Given any two of the three Woburn colours in no particular order, your task is to name the third one.

Input Format

There will be two lines of input given to your program. Each line will contain one of Woburn's colours in all uppercase letters.
Each of the colours will be one of "`RED`", "`BLUE`", or "`WHITE`" (without quotation marks), and the two colours given are guaranteed to be distinct.

Output Format

Output a single line containing the Woburn colour that was not mentioned in the input. Your output must be in all uppercase.

Sample Input

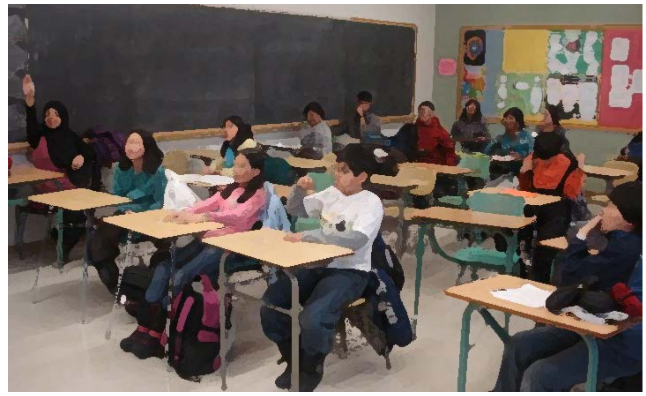
```
RED
BLUE
```

Sample Output

```
WHITE
```

Problem J2: Grouping Recruits

20 Points / Time Limit: 2.00s / Memory Limit: 16M *Submit online: wcipeg.com/problem/wc151j2*



Now that you know a little more about Woburn, let's learn a bit more about the amazing computer science enrichment opportunity that it holds. You might not know that every couple of years, Woburn's very own Programming Enrichment Group recruits elementary and middle school students who aspire to be talented little computer scientists. Indeed, training programmers from such a young age has been a key principle behind the club's success.

Since the number of recruits is usually very high, the PEG leader(s) will often have trouble handling them alone and all at once. Thus, an age-old solution has been devised – splitting them up into groups, each of which will be taught by a senior member of the club! However, this is a more challenging task than it seems, since senior members tend to complain amongst themselves about how unfair it is that some of them have to deal with teaching significantly more recruits than others.

Let's say that there are N recruits and M senior members within the club ($1 \le M \le N \le 100$). We would like to "evenly" divide the N recruits up into M groups such that the numbers of recruits the senior members have to handle are roughly equal to each other. More precisely, we would like to split up the recruits into M groups such that the difference between the size of the largest group and the size of the smallest group is as small as possible. Only this way can we prevent certain senior members from complaining that their duty is too hard and unfair compared to other senior members.

Please help PEG divide the recruits into "fair" groups.

Input Format

The first line of input will contain a single integer N representing the number of recruits to be split up.
The second line of input will contain a single integer M representing the number of groups that must be made.

Output Format

Output one line for each group size in the format "`X group(s) of Y`" where Y is the size of the group and X is the number of groups with this size in your grouping scheme.
You may output the lines in any order. Consult the sample outputs if you need further clarification of the format.

Sample Input 1

```
30
3
```

Sample Output 1

```
3 group(s) of 10
```

Explanation 1

There are 30 recruits and three senior members. We can just do three nice groups of 10 recruits each.

Sample Input 2

```
21
5
```

Sample Output 2

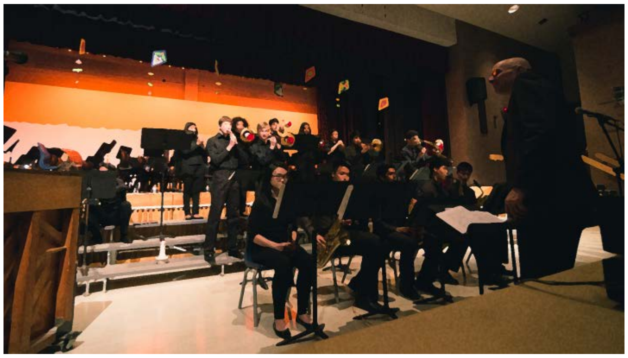
```
4 group(s) of 4
1 group(s) of 5
```

Explanation 2

There are 21 recruits and five senior members. The fairest option would be to create four groups of 4 and one group of 5 (where the difference between the smallest and largest group is $5 - 4 = 1$).
This is more "fair" than, for instance, creating four groups of 5 and one group of 1 (where the difference between the largest and smallest group sizes is $5 - 1 = 4$).

Problem J3: Jazz Concert

20 Points / Time Limit: 2.00s / Memory Limit: 16M *Submit online: wcipeg.com/problem/wc151j3*



The renowned Woburn Jazz Band is putting on a concert!

Their current repertoire consists of N ($2 \le N \le 100$) different songs that are numbered with distinct integers from 1 to N. The i-th of these songs (for i = 1..N) has a duration of T_i ($1 \le T_i \le 100$) minutes.

Since the band is so spectacular, they are always receiving requests to fill more and more time. Unfortunately, rehearsal times are tight and practicing any more than these N pieces is simply not feasible. Through Mr. Farrow's decades of experience as conductor of the band, he has observed that they can always get away with repeating at most two of their N songs to extend the concert (surely the audience won't notice). With that in mind, the band has decided to pick two songs to play twice, while each of the $N - 2$ remaining songs will be played exactly once. Furthermore, there will always be a 10-minute intermission in the middle of the concert.

With all this information, Mr. Farrow needs to know ahead of time the length of the entire concert so that he can report it to the principal for scheduling purposes. Given the lengths of all the pieces in the Jazz Band's repertoire, please help Mr. Farrow determine the maximum amount of time (in minutes) that the concert could possibly last, including the intermission time and given that optimal choices will be made for which two songs are to be repeated.

Input Format

Line 1 of input will contain a single integer N, representing the number of songs in the band's repertoire.
N lines will follow, with the i-th of these lines (for i = 1..N) containing a single integer T_i, representing the duration of the i-th song in minutes.

Output Format

Output a single integer representing the maximum possible length of the concert in minutes.

Sample Input

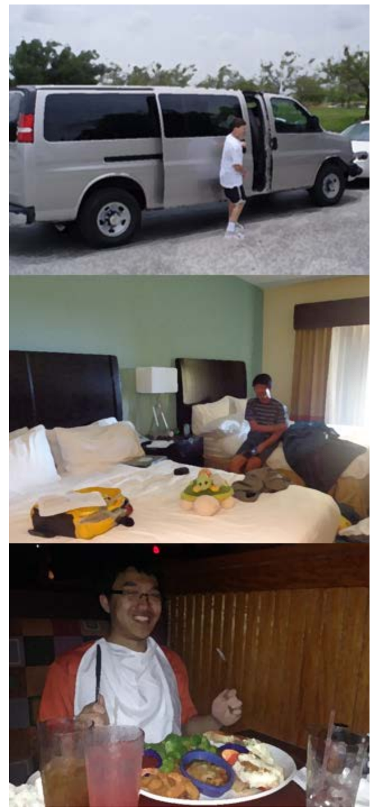
```
4
10
10
20
25
```

Sample Output

```
120
```

Problem J4: Trip Budgeting

50 Points / Time Limit: 2.00s / Memory Limit: 16M *Submit online: wcipeg.com/problem/wc151j4*



The PEG trip at the end of the school year is a long-lasting tradition of the club. The leaders usually decide on three teams (junior, intermediate, and senior) based on whom they find to be most attentive and hard-working throughout the years. Then, the deserving teams are taken to some place in the USA (a different city every year!) where they will compete, and most probably dominate, in the American Computer Science League (ACSL) All-Star competition. The trip will then consist of several action-packed days of competing, dining, frolicking, and touring the local attractions. While most PEG students who are selected to go are simple enlightened by the fantastic opportunity, few of them are aware of the rigorous planning and budgeting that takes place behind-the-scenes.

Budgeting is a complex task. We must account for transportation, hotel, food, and most importantly – attractions. Choices must be made in each of these categories with consideration for how much money is going to be spent, and how exciting the trip is expected to be. To make each trip memorable enough for the PEG history books, the PEG leaders would like this year's trip to have at least an excitement level of E_{min} ($1 \le E_{min} \le 10^9$) units. Funding has always been a complicated matter due to Woburn's Student Activity Council (SAC). The SAC is willing to help the club satisfy its minimal excitement level, provided that it is done in the cheapest way possible. The cost and excitement of the trip will be calculated with the following rules:

- The base costs of transportation, hotel, and food in dollars are respectively T_{base}, H_{base}, and F_{base} ($1 \le T_{base}, H_{base}, F_{base} \le 10^7$).
- There are N ($1 \le N \le 20$) attractions numbered from 1 to N. The team can attend each attraction at most once.
  - Each attraction has an E_i, T_i, H_i, and F_i value associated with it ($0 \le E_i, T_i, H_i, F_i \le 10^7$).
  - If attended, the i-th attraction (for $i = 1..N$) will contribute E_i units of excitement to the overall trip.
  - By going to a particular attraction, money may be saved or extra money may be spent. For instance, the team might have to pay extra transportation costs to reach the event. The team might also have to pick a more expensive hotel in order to be close enough to attend it. Conversely, many attractions provide free food, so the team might actually save money on food.
  - By choosing attraction i, the team must be prepared to also pay an additional T_i dollars for transportation on top of T_{base}. That is, the *overall transportation cost* is the sum of the base transportation cost and the transportation costs incurred across all of the attractions they decide to visit.
  - Each attraction has an associated hotel nearby that costs H_i dollars to stay at. Since the team will not be attending more than one hotel, we have to prepare for the worst and assume that the *overall hotel cost* for the entire trip will be the maximum of H_{base} and H_i across all of the attractions that the team eventually decides to attend.
  - By choosing attraction i, the team will actually *save* F_i dollars on food because it will be provided at the venue. That is, the *overall food cost* will be the base food cost F_{base} subtract the F_i for each attraction i that was attended. The base food cost cannot go below zero – after F_{base} has been reduced to zero, attending additional attractions will have no effect on the nonexistent overall food cost.

- The total cost of the trip is equal to the sum of the overall costs of transportation, hotel, and food respectively.
- The excitement of the trip is equal to the sum of E_i across all of the attractions that the team decides to attend.

Given this, please help the PEG leaders determine the minimum possible total cost that will produce at least E_{min} units of excitement for the overall trip. It is guaranteed that this is possible.

Input Format

Line 1 will contain four space-separated integers in the following order: E_{min}, T_{base}, H_{base}, and F_{base}.
Line 2 will contain a single integer N.
There will be N lines to follow. The i-th of these lines (for $i = 1..N$) will contain four space-separated integers: E_i, T_i, H_i, and F_i.

Output Format

Output a single integer – the minimum cost (in dollars) required to achieve at least E_{min} units of excitement on the trip.

Sample Input

```
50 2000 20000 4000
5
30 300 20000 400
40 500 30000 2000
10 100 15000 800
30 500 11000 1000
20 400 20000 500
```

Sample Output

```
24600
```

Explanation

The team wants to achieve a minimum of 50 units of excitement, and has given the five attractions with excitement ratings of 30, 40, 10, 30, and 20 respectively. By choosing every attraction except for number 2, the team is actually able to achieve 90 units of excitement at a cost of 24600 dollars. The costs are calculated as follows:

- Transportation costs 2000 + 300 + 100 + 500 + 400 = 3300 dollars.
- The most expensive hotel across all of the attractions being attended costs 20000 dollars.
- Food was original going to cost 4000 dollars, but 400 + 800 + 1000 + 500 = 2700 dollars were saved by eating at the venues of the attractions.

Thus, the total cost of the trip is 3300 + 20000 + (4000 − 2700) = 24600 dollars. In fact, this is the best possible answer.