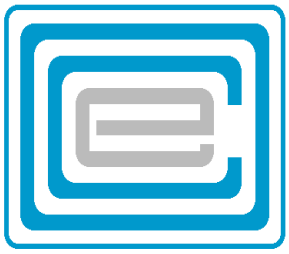

ECOO 2018

Programming Contest Questions

Provincial Competition (Round 3)

May 12, 2018

Problem 1: Balanced

Ms. Daisy teaches a class of **B** boys and **G** girls that need to line up every morning to take attendance. Ms. Daisy thinks that a line is "balanced" if at least one of the boys is equidistant from two of the girls in the line. For example, a *girl-boy-boy-girl* line is not balanced because both boys are closer to one of the girls, but a *girl-girl-boy-boy-girl* line is balanced because the first boy is equidistant from the first and last girls.

Ms. Daisy likes it when the students form a balanced line. Can you help her figure out the number of balanced lines that the students can form? Two lines are considered distinct if at least one student has a different position in each line.

Input Specifications

DATA11.txt (DATA12.txt for the second try) will contain 10 datasets. Each dataset contains two integers **B, G** ($1 \le$ **B, G** $\le 1{,}000{,}000$).

For the first four datasets, **B+G** ≤ 20.

Output Specifications

For each data set, output the number of balanced lines that can be formed, modulo 1,000,000,007.
Note: A Modulo B is the remainder of A/B.

Sample Input (Three Datasets Shown)	Sample Output
1 2 2 2 3 2	2 8 48

Explanation of Sample Input

In the first case, a balanced line must have a girl, then the boy, and then the other girl. Either girl can come first, which gives us two balanced lines.
In the second case, a balanced line has either a *boy-girl-boy-girl* pattern or a *girl-boy-girl-boy* pattern.
In the third case, an example balanced line would have a *girl-boy-boy-boy-girl* pattern (the boy in the middle is equidistant from the two girls).

Problem 2: Vegenère

The Vegenère cipher is a famous cipher which uses a keyword to shift the value of each character of the message by some amount. For example, if the message is “HELLOWORLD”, and the keyword is “CODE”, then the Vegenère cipher first repeats the keyword until its the same length as the message, then encrypts the message by “adding” the two strings together:

```
  HELLOWORLD
+ CODECODECO
= KTPQRLSWOS
```

To add the strings together, we “shift” each character in the message by the character in the keyword. For example, shifting ‘Y’ by ‘A’ returns ‘Z’, shifting by ‘B’ returns ‘A’, shifting by ‘C’ returns ‘B’, and so on. Note that the shift wraps around the alphabet, i.e. ‘A’ is after ‘Z’.

Alice and Bob have been spreading rumours behind Eve’s back. They do not want Eve to find out, so they have encrypted their messages using the Vegenère cipher. However, they have made a fatal flaw by choosing an English word as their keyword and only sending messages consisting of English words. Can you help Eve decipher what they are saying?

Input Specifications

DATA21.txt (DATA22.txt for the second try) will contain 10 datasets. Each dataset begins with an integer **N** (1 ≤ **N** ≤ 100,000), the number of words in the English dictionary that Alice and Bob are using.
N lines follow, each containing a word in the English language. The total size of the dictionary will be at most 1,000,000 characters.
The next line contains the encrypted message, which will be at most 100,000 characters in length.
All strings in the input will consist of upper-case English characters.

For 30% of the cases, **N** ≤ 10 and the encrypted string will contain at most two words.
For 60% of the cases, **N** ≤ 100.

Output Specifications

For each dataset, output the first ten characters of the decrypted message and the sum of the squared lengths of the words in the message. If there are multiple solutions, output the one that comes first alphabetically. If there are multiple interpretations of a message, output the one that minimizes the sum of squared lengths of the words in the message.

(Sample datasets on next page)

Sample Input (Two Datasets Shown)

```
3
CODE
HELLO
WORLD
KTPQRLSWOS
5
EVE
IS
VERY
COOL
SILLY
HKTUVKTDBHXXON
```

Sample Output

```
HELLOWORLD 50
EVEISVERYS 54
```

Problem 3: Currency Exchange

Eduardo is planning for a trip to Mexico this summer and he needs to buy some pesos for his trip. Eduardo doesn't trust currency exchange businesses as he believes they give bad rates, so instead he will ask his friends to trade currencies.

Eduardo has **M** friends that are willing to trade currencies. Each friend is willing to trade some currency that they have for some other currency that they want. Also, each friend has their own exchange rate and is willing to trade any amount of currency (even fractional amounts).

Eduardo currently has **D** Canadian dollars and wants to get as many pesos as possible so that he can fully enjoy his trip. Can you help Eduardo figure out how many pesos he can get?

Input Specifications

DATA31.txt (DATA32.txt for the second try) will contain 10 datasets. Each dataset begins with three integers **N**, **M, D** ($2 \le N \le 5000$, $1 \le M, D \le 5000$): the number of available currencies, the number of friends willing to trade currency, and how many dollars Eduardo currently has, respectively. The Canadian dollar has index 1, and the peso has index N.

The next **M** lines each describe the trades that Eduardo's friends are willing to perform. Each line contains three integers **A**, **B** ($1 \le A, B \le N$) and one positive real number **R** ($0.1 \le R \le 10$) given to four decimal places, where:

- **A** is the index of the currency that the friend wants.
- **B** is the index of the currency that the friend has.
- **R** is the exchange rate that the friend set. The friend will sell 1 unit of currency **B** for **R** units of currency **A**.

For the first four cases, each exchange rate **R** is greater than or equal to 1.

Output Specifications

For each dataset, output the maximum number of pesos that Eduardo can get, rounded to two decimal places. If Eduardo can get more than one billion pesos, output "Billionaire!" instead. It is guaranteed that the answer will not be within 1000 pesos of one billion.

Sample Input (Two Datasets Shown)

```
3 3 3
1 3 1.0000
1 2 2.0000
2 3 0.3000
4 4 1
1 2 0.5000
2 3 0.5000
3 4 0.5000
4 1 7.0000
```

Sample Output

```
5.00
Billionaire!
```

Problem 4: Circular Cities

In Ringworld, there are **M** cities numbered from 1 to **M** which lie uniformly spaced along a circular, two-way road. It takes one hour to travel between city **i** and city **i+1** for $1 \le i < M$ and one hour to travel between city 1 and city **M**.

There are **N** students and **N** teachers living in the cities. You would like to assign every teacher to a distinct student such that the sum of the times needed for each student to travel to their assigned teacher's city is minimized.

What is the minimum sum of the travel times for a teacher-student assignment?

Input Specifications

DATA41.txt (DATA42.txt for the second try) will contain 10 datasets. Each dataset begins with two integers **N** and **M** ($1 \le N \le 500{,}000$, $1 \le M \le 10^9$).

The next line contains **N** integers A_i ($1 \le A_i \le M$) such that student **i** lives in city A_i. The third line contains **N** integers B_i ($1 \le B_i \le M$) such that teacher **i** lives in city B_i. At most one student or teacher lives in each city.

For the first 4 cases, $N \le 1000$.

Output Specifications

For each dataset, output the minimum sum of the travel times for a teacher-student assignment.

Sample Input (Two Datasets Shown)

```
2 5
1 4
5 3
5 100
10 20 30 40 50
60 70 80 90 100
```

Sample Output

```
2
130
```


ECOO 2018 Question Development Team

John Ketelaars ..... ECOO-CS Communications
Stella Lau ..... University of Cambridge
Davin Beharry ..... Ryerson University
Sam Scott ..... Mohawk College
Andrew Seidel ..... John Fraser Secondary School, Mississauga
David Stermole ..... ECOO-CS President
Reyno Tilikaynen ..... University of Waterloo
Andy Huang ..... University of Waterloo