TCS CodeVita24

Check out my Youtube and Instagram

Most Important Topics

- 2 Pointers Approach
- Searching Sorting
- Number Theory
- Stacks and Queues
- Binary Search
- Dynamic Programming
- Graphs
- Trees
- Recursion
- Segment
- Heaps

Resources:

- https://www.youtube.com/@abdul_bari **(Graphs + Trees)**
- https://www.youtube.com/watch?v=tyB0ztf0DNY&list=PLgUwDviBIf0pwFf-BnpkXxs0Ra0eU2sJY **(DP)**
- https://www.geeksforgeeks.org/top-50-tree-coding-problems-for-interviews/ **(Tree)**
- https://www.youtube.com/playlist?list=PLgUwDviBIf0rAuz8tVcM0AymmhTRsfaLU **(LinkedList)**

PREVIOUS YEAR QUESTION

Problem 1 : Date Time

Arun and his sister Usha are challenging each other with some mathematical puzzles. Usha, the cleverer one, has come up with the idea of Giving Arun 12 distinct digits from 0 to 9, and have him from the largest date time in 2018 with them. Arun is a little nervous, and asks you to help him with a computer program.
Usha will give Arun 12 distinct digits. He needs to create a date time combination in the year 2018: the date in the MM/DD form (all four digits must be present), and the time in the format HH:MM (all four digits must be present). The date may be from 01/01 to 12/31 and the time may be from 00:00 to
23:59 (in the 24 hour format).
The digits provided may be used only once in the answer that Arun gives.
If more than one date time combination may be formed, Arun needs to give the latest valid date time possible in the year 2018.
Constraints
Single digits (any of 0-9)

Input Format

A line consisting of a sequence of 12 (not necessarily distinct) single digits (any of 0-9) separated by commas. The sequence will be non-decreasing.

Output

The maximum possible valid date time in the year 2018. The output must be in the format
MM/DD HH:MM
If no date time can be constructed, the output should be 0.

Example1 :

Input 0,0,1,2,2,2,3,5,9,9,9,9
Output 12/30 22:59

Explanation:

The 12 digits to be used by Arun are given.
The maximum valid date time using only the digits given, and with each digit used at most once is 12/30 22:59
This is the output.

Example 2

Input: 3,3,3,3,3,3,3,3,3,3,3,3
Output: 0

Explanation

As no digit less than 3 is present in the input, a valid month cannot be formed. Hence no valid Date time can be formed with the input digits.

Problem 2 : Digital Time

The objective is to form the maximum possible time in the HH:MM:SS format using any six of nine given single digits (not
necessarily distinct)
Given a set of nine single (not necessarily distinct) digits, say 0, 0, 1, 3, 4, 6, 7, 8, 9, it is possible to form many distinct times
in a 24 hour time format HH:MM:SS, such as 17:36:40 or 10:30:41 by using each of the digits only once. The objective is to
find the maximum possible valid time (00:00:01 to 24:00:00) that can be formed using some six of the nine digits exactly
once. In this case, it is 19:48:37.

Input : A line consisting of a sequence of 9 (not necessarily distinct) single digits (any of 0-9) separated by commas. The sequence will
be non-decreasing

Output : The maximum possible time in a 24 hour clock (00:00:01 to 24:00:00) in a HH:MM:SS form that can be formed by using some
six of the nine given digits (in any order) precisely once each. If no combination of any six digits will form a valid time, the
output should be the word Impossible

Example 1
Input: 0,0,1,1,3,5,6,7,7
Output: 17:57:36

Explanation:

The maximum valid time in a 24 hour clock that can be formed using some six of the 9 digits precisely once is 17:57:36

Example 2

Input: 3,3,3,3,3,3,3,3,3
Output: Impossible

Explanation:

No set of six digits from the input may be used to form a valid time.

Problem3 : Greedy Hostel Owner

You know that summers are at peak this year and every day is hot and due to this everyone is using coolers and ACs and a lot of electricity is consumed by the people. You are living in a hostel and your hostel owner decided to charge extra for electricity consumption. To achieve this he put one separate electricity meter for every room and connected all those meters to central meter.
But the hostel owner is a bit greedy and wants to manipulate the meters to show a reading that is more than the actual
consumption of electricity. He also encrypted all the meters with alphabets. The technique he used for encrypting is as follows:
Every meter has 6 Alphabets i.e. 6 digits.
Every alphabet is in upper case.
Allowed alphabets are A, B, C, D, E, F, G, H, I, J.
A corresponds to 0, B = 1 and similarly C = 2, D = 3, E = 4, F = 5, G = 6, H = 7, I = 8, J = 9
The interpretation rules change as follows:
If the alphabet next to J is A, then J represents 0. Similarly, if the alphabet after I is B, then I counts as 1 (and not 8), the alphabet after H is C, then H represents 2. The same is true if D follows G and if E follows F. Note that A, B, C, D and E will
always retain their respective values. When J is not followed by A, J will represent 9 and similar rules for I, H, G and F
You are given central meter reading and encrypted readings of all the meters in the hostel. Your task is to find out whether the
the owner is Greedy or Innocent. If he is greedy then print the unit difference otherwise print innocent.
Owner is greedy if and only if

(units of all meters in the hostel except central meter < central meter units)
Input Format:
First line contains an integer N, giving the number of rooms in the hostel.
The next line contains N strings each of length 6 characters giving the readings of the meters in the rooms
The next line contains an integer that gives the reading in the central meter

Output Format:

First line containing either GREEDY or INNOCENT
If the first line is GREEDY, the next line should contain the difference (as a decimal number) between the central meter reading
and the consumption shown in the rooms.

Constraints:

Number of rooms <= 100

Example 1

Input
3
JAABHF JAACJA JAACDA
500
Output
GREEDY
105
A B C D E F G H

Example 2

Input
8
JAACJA JAABCH JAABHD JAACAF JAJAJJ JAABEJ JAACJJ JAACDI
1500
Output

INNOCENT

Explanation

The readings are,
000200, 000127, 000173, 000205, 0000099, 000149, 000299, 000238
The sum of these readings is 1490 < 1500, the central meter reading. Hence the owner is INNOCENT.

Problem 4: Civil War

In this superhero epic, the denizens of the Marvel Universe are forced to pick sides when Captain America and Iron Man come to blows over ideological differences.
The government decides to push for the Hero Registration Act, a law that limits a hero's actions. This results in a division in The Avengers. Iron Man stands with this Act, claiming that their actions must be kept in check otherwise cities will continue to be destroyed, but Captain America feels that saving the world is daring enough and that they cannot rely on the government to protect the world. And here the civil war begins.

They are trying make their team stronger by adding more avengers to their team. There are N avengers lined up.

Rules to add avenger to their team-

- Any team can start first. But they will alternatively only.
- They can select avenger from any side. But if they start from one side they can't move to other side in current chance.
- They can select consecutive avengers as many they want.
- They will stop only when all the avengers are part of either side.
- Every Avenger has a power associated with him
- There are some spurious avengers who will decrease the overall power of the team.

Both teams will select players optimally. Find the difference of powers of the two teams

Constraints
1<= N <= 10^6
-10^9 <= p[i] <= 10^9

Input
First line contains an integer denoting the number of Avengers(N).
Next lines contain N space separated values denoting power of every avenger(P[i]).

Output
Print the difference of the powers of teams
– Time Limit (secs)
1

Examples :
Input
5
2-78-1 20

Output
2

Problem 5: Possible Legal Subsets

You are given N comma-separated Strings. You need to form all possible legal subsets of these N strings. These subsets will be a combination of zero or more of these N Strings After forming the subsets, they will be ranked in a particular onder. The legal subset formation and ranking logic is as described below

- Rank 1 will always be an empty set

- Next N ranks will be the N Strings that appear in the order they are provided in the input

- After N + 1 ranks, you need to combine N strings such that all legal combinations are formed

- Legal combination simply means that while combinations are formed, the string that appears to the left of a particular string in the input, can never appear to the right of that particular string, when subsets are formed

- A subset with less elements will be ranked higher than a subset with more elements (NOTE-Rank 1 is higher than rank 2)

- Refer Example 2 to get a better understanding of how subsets are formed and ranked

- It is guaranteed that

- N>=1

- All N strings are unique

Example: you are having an input string “aa,cc,bb” in this string we can see we have three strings which are comma separated. Now from this group of string we have to create all possible subset of strings. 8 subsets can be formed from these strings. And they are as follows:

1. {}
2. {aa}
3. {cc}
4. {bb}
5. {aa,}

Note: here we can see the ranks given to the subsets are first by size i.e., the subset with lesser number of strings is ranked higher than the subset with higher size. If the subsets have equal number of strings then, the combination which is formed earlier (by virtue of combining strings in order they appear in input), gets a higher rank.

For example, rank of su bset (aa,cc) > rank of (aa,bb) because string cc is appearing prior to string bb in the input. Similarly, rank of (cc) > rank of (bb).

You are provided one rank R and for that you have to print the Rth subset from all legal subsets.

Constraints:

0<N<=10^2

0<R<=10^18

Input

First line contains an integer N which is number of strings in group.

Second line contains an integer R, for which you have to find Rth subset from all legal subsets.

Third line contains N comma-separated strings basis which the subsets should be formed

Output:

From all possible legal subsets find the subset whose rank is R

Time Limit (secs)
1

Input

2

4

a,b

Output

a,b

Explanation:

Given that N = 2, given

Second line: Rank to be find: 4th

Third line: Given group of strings: a,b

Possible subsets & Rank

{}-1

{a} -2

{b}-3

{a, b}-4

Output – a,b (4th rank corresponds to a,b)

Problem 6: Seating Arrangement

You are a caretaker of a waiting room and you have to take care of empty seats such that all the people should sit together. Imagine the seats are in a straight line like in a movie theatre. People are seated on random seats initially. Your task is to make them sit together so that minimum number of people change their position. Also, they can be made to sit together in many ways. Find the number of ways you can make them sit together by requiring only minimal people movement.

“E” depicts an empty seat and “O” depicts an occupied seat. Input will be given in the form of a string.

Example: OEOEO

As we can see, only seat number 1, 3, 5 are occupied and 2 and 4 are empty.

Case 1: If we move 5th person to 2nd position, they can all be together with only one person moving his/her place.

Case 2: If we movement 1st person to 4th position, they can all be together with only one person moving his/her place.

They can all be together with only one movement and this can be done in 2 ways. Print the minimum number of movements required and the number of ways this minimum movement can help achieve the objective.

Note: If they are already sitting together, Print “00” as output.

Constraints
0 <N <= 100000

Input

First line contains an integer N which depicts the number of seats

Second line contains N characters each of which are either “O” or “E”. “O” denotes an occupied seat and “E” denotes an empty seat.

Output
Print minimum number of movements required and the number of ways in which all people can be made to sit together without exceeding minimum number of movements by space
Time Limit (secs)
1

Examples
Input
5

OEOEO

Output
1 2

Explanation:

Given data of 5 seats in the queue,

Seat number 2 and 4 are unoccupied and all the other seats are occupied.

We can make them sit together by moving only one person near to the other. It can be done in 2 ways:

OOOEE (Moving 4 person to 2º position)

EE000 (Moving 1 person to 4 position)

Problem 7: Polygon with Maximum Area

You are given N number of coordinates and you have to create a polygon from these points such that they will make a polygon with maximum area.

Note: coordinates provided in the input may or may not be in sequential form.

Constraints
1 <= N <= 10

Input:

First line contains an integer N which depicts number of co-ordinates

Next N lines consist of two space separated integer depicting coordinates of in form of xy

Output:
Print the maximum possible area possible by creating a polygon by joining the coordinates.

If the area is in decimal form, print the absolute value as output.

Time Limit (secs): 1

Examples:

Input:

4

0 0

2 0

0 2

2 2

Output: 4

Explanation:

As we can imagine that these points will make a square shape and the maximum possible area made by the polygon will be 4.

Problem 8: Consecutive Prime Sum

Question – : Some prime numbers can be expressed as a sum of other consecutive prime numbers.

- **For example**
    - 5 = 2 + 3,
    - 17 = 2 + 3 + 5 + 7,
    - 41 = 2 + 3 + 5 + 7 + 11 + 13.

      Your task is to find out how many prime numbers which satisfy this property are present in the range 3 to N subject to a constraint that summation should always start with number 2.

Write code to find out the number of prime numbers that satisfy the above-mentioned property in a given range.

Input Format: First line contains a number N

Output Format: Print the total number of all such prime numbers which are less than or equal to N.

Constraints: 2<N<=12,000,000,000

Problem 9 Counting Rock Sample

Question – : Juan Marquinho is a geologist and he needs to count rock samples in order to send it to a chemical laboratory. He has a problem: The laboratory only accepts rock samples by a range of its size in ppm (parts per million).

Juan Marquinho receives the rock samples one by one and he classifies the rock samples according to the range of the laboratory. This process is very hard because the number of rock samples may be in millions.

Juan Marquinho needs your help, your task is to develop a program to get the number of rocks in each of the ranges accepted by the laboratory.

Input Format: An positive integer S (the number of rock samples) separated by a blank space, and a positive integer R (the number of ranges of the laboratory); A list of the sizes of S samples (in ppm), as positive integers separated by space R lines where the ith line containing two positive integers, space separated, indicating the minimum size and maximum size respectively of the ith range.

Output Format: R lines where the ith line contains a single non-negative integer indicating the number of the samples which lie in the ith range.

Constraints:

- 10 <= S <= 10000

- 1 <= R <= 1000000
- 1<=size of Sample <= 1000

Example 1

- Input: 10 2
- 345 604 321 433 704 470 808 718 517 811
- 300 350
- 400 700

Output: 2 4

Explanation:

There are 10 samples (S) and 2 ranges (R). The samples are 345, 604,811. The ranges are 300-350 and 400-700. There are 2 samples in the first range (345 and 321) and 4 samples in the second range (604, 433, 470, 517). Hence the two lines of the output are 2 and 4

Problem 10 : kth Largest factor of N

Question -: A positive integer d is said to be a factor of another positive integer N if when N is divided by d, the remainder obtained is zero. For example, for number 12, there are 6 factors 1, 2, 3, 4, 6, 12. Every positive integer k has at least two factors, 1 and the number k itself.Given two positive integers N and k, write a program to print the kth largest factor of N.

Input Format: The input is a comma-separated list of positive integer pairs (N, k).

Output Format: The k^{th} highest factor of N. If N does not have k factors, the output should be 1.

Constraints:

- 1<N<10000000000
- 1<k<600.

You can assume that N will have no prime factors which are larger than 13.

Example 1

- **Input**: 12,3
- **Output**: 4

Explanation: N is 12, k is 3. The factors of 12 are (1,2,3,4,6,12). The highest factor is 12 and the third largest factor is 4. The output must be 4.

Example 2

- **Input**: 30,9
- **Output**: 1

Explanation: N is 30, k is 9. The factors of 30 are (1,2,3,5,6,10,15,30). There are only 8 factors. As k is more than the number of factors, the output is 1.

Problem 11: String Pair

Problem Description

One person hands over the list of digits to Mr. String, But Mr. String understands only strings. Within strings also he understands only vowels. Mr. String needs your help to find the total number of pairs which add up to a certain digit D.The rules to calculate digit D are as follow :-

Take all digits and convert them into their textual representation.

Next, sum up the number of vowels i.e. {a, e, i, o, u} from all textual representation.

This sum is digit D

Now, once digit D is known find out all unordered pairs of numbers in input whose sum is equal to D. Refer example section for better understanding.

Constraints

1 <= N <= 100

1 <= value of each element in second line of input <= 100

Number 100, if and when it appears in input should be converted to textual representation as hundred and not as one hundred. Hence number of vowels in number 100 should be 2 and not 4

Input

First line contains an integer N which represents number of elements to be processed as input

Second line contains N numbers separated by space

Output

Lower case representation of textual representation of number of pairs in input that sum up to digit D

Note: – (If the count exceeds 100 print “greater 100”)

Examples

Input : 5

1 2 3 4 5

Output : one

Input : 3

7 4 2

Output : zero

Question 12: Elections

Elections are going on, and there are two candidates A and B, contesting with each other. There is a queue of voters and in this queue some of them are supporters of A and some of them are supporters of B. Many of them are neutral. The fate of the election will be decided on which side the neutral voters vote. Supporters of A and supporters of B make attempt to win the votes of neutral voters.

The way this can be done is explained below:

1. The voter queue is denoted by three characters, viz {-, A, B}. The – denotes neutral candidate, A denotes supporter of candidate A and B denotes supporter of candidate B.

2. Supporters of A can only move towards the left side of the queue.

3. Supporters of B can only move towards the right side of the queue.

4. Since time is critical, supporters of both A and B will move simultaneously.

5. They both will try and influence the neutral voters by moving in their direction in the queue. If supporter of A reaches the neutral voter before supporter of B reaches him, then that neutral voter will become a supporter of candidate A.

6. Similarly, if supporter of B reaches the neutral voter before supporter of A reaches him, then that neutral voter will become a supporter of candidate B.

7. Finally, if both reach at the same time, the voter will remain neutral. A neutral vote cannot decide the outcome of the election.

8. If finally, the queue has more votes for candidate A, then A wins the election. If B has more votes, then B wins that election. If both have equal votes, then it will be a coalition government.

Refer Examples section for understanding the dynamics of how the supporters influence the neutral voters.

Your task is to find the outcome of the election.

Note: There are no test cases where all votes are neutral.

Input

First line contains an integer which is length of queue of voters.

Second line contains characters {-, A, B}, in which denotes

· A = voter who is supporter of candidate A

· B = voter who is supporter of candidate B

· – = neutral voter

Output

Print candidate with maximum number of votes. If they have equal number of votes, print “Coalition government“.

Examples

Input :14

–AB–AB—A–

Output : A

Input : 4

A—

Output : A

Problem 13: Constellation

Three characters { #, *, . } represents a constellation of stars and galaxies in space. Each galaxy is demarcated by # characters. There can be one or many stars in a given galaxy. Stars can only be in the shape of vowels { A, E, I, O, U }. A collection of * in the shape of the vowels is a star. A star is contained in a 3x3 block. Stars cannot be overlapping. The dot(.) character denotes empty space.

Given 3xN matrix comprising of { #, *, . } character, find the galaxy and stars within them.

Note: Please pay attention to how vowel A is denoted in a 3x3 block in the examples section below.

Constraints

3 <= N <= 10^5

Input

Input consists of a single integer N denoting the number of columns.

Output

The output contains vowels (stars) in order of their occurrence within the given galaxy. The galaxy itself is represented by the # character.

Example 1

Input

18

* . * # * * * # * * * # * * * . * .

* . * # * . * # . * . # * * * * * *

* * * # * * * # * * * # * * * * . *

Output

U#O#I#EA

Explanation

As can be seen, the stars make the image of the alphabets U, O, I, E, and A respectively.

Problem 14: Prime Time Again

Here on earth, our 24-hour day is composed of two parts, each of 12 hours. Each hour in each part has a corresponding hour in the other part separated by 12 hours: the hour essentially measures the duration since the start of the daypart. For example, 1 hour in the first part of the day is equivalent to 13, which is 1 hour into the second part of the day.
Now, consider the equivalent hours that are both prime numbers. We have 3 such instances for a 24-hour 2-part day:

5~17
7~19
11~23

Accept two natural numbers D, P >1 corresponding respectively to several hours per day and the number of parts in a day separated by a space. D should be divisible by P, meaning that the number of hours per part (D/P) should be a natural number. Calculate the number of instances of equivalent prime hours. Output zero if there is no such instance. Note that we require each equivalent hour in each part of a day to be a prime number.

Example:

Input: 24 2
Output: 3 (We have 3 instances of equivalent prime hours: 5~17, 7~19, and 11~23.)

Constraints

10 <= D < 500

2 <= P < 50

Input

The single line consists of two space-separated integers, D and P corresponding to the number of. hours per day and number of parts in a day respectively

Output

Output must be a single number, corresponding to the number of instances of equivalent prime number, as described above

Example 1

Input
36 3

Output
2

Explanation

In the given test case D = 36 and P = 3
Duration of each daypart = 12
2~14~X
3~15~X
5~17~29 - an instance of equivalent prime hours
7~19~31 - an instance of equivalent prime hours
11~23~X
Hence the answer is 2.

Problem 15: Minimum Gifts

A Company has decided to give some gifts to all of its employees. For that, the company has given some rank to each employee. Based on that rank, the company has made certain rules for distributing the gifts.

The rules for distributing the gifts are:

Each employee must receive at least one gift.

Employees having higher ranking get a greater number of gifts than their neighbours.

What is the minimum number of gifts required by the company?

Constraints

1 < T < 10

1 < N < 100000

1 < Rank < 10^9

Input

The first line contains integer T, denoting the number of test cases.

For each test case:

The first line contains integer N, denoting the number of employees.

The second line contains N space-separated integers, denoting the rank of each employee.

Output

For each test case print the number of minimum gifts required on a new line.

Input

2

5

1 2 1 5 2

2

1 2

Output

7

3

Explanation

For test case 1, adhering to the rules mentioned above,

Employee # 1 whose rank is 1 gets one gift

Employee # 2 whose rank is 2 gets two gifts

Employee # 3 whose rank is 1 gets one gift

Employee # 4 whose rank is 5 gets two gifts

Employee # 5 whose rank is 2 gets one gift

Therefore, total gifts required is 1 + 2 + 1 + 2 + 1 = 7

Similarly, for test case 2, adhering to the rules mentioned above,

Employee # 1 whose rank is 1 gets one gift

Employee # 2 whose rank is 2 gets two gifts

Therefore, the total gifts required is 1 + 2 =

Problem 16 :Minimize the sum

Given an array of integers, perform at most K operations so that the sum of elements of a final array is minimum. An operation is defined as follows -

Consider any 1 element from the array, arr[i].

Replace arr[i] by floor(arr[i]/2).

Perform the next operations on the updated array.

The task is to minimize the sum after utmost K operations.

Constraints

1 <= N, K <= 10^5.

Input

The first line contains two integers N and K representing the size of the array and the maximum number of operations that can be performed on the array respectively.

The second line contains N space-separated integers denoting the elements of the array, arr.

Output

Print a single integer denoting the minimum sum of the final array.

Input

4 3

20 7 5 4

Output

17

Explanation

Operation 1 -> Select 20. Replace it with 10. New array = [10, 7, 5, 4]

Operation 2 -> Select 10. Replace it with 5. New array = [5, 7, 5, 4].

Operation 3 -> Select 7. Replace it with 3. New array = [5,3,5,4].

Sum = 17.

Problem 17 :Railway Station

Given the schedule of trains and their stoppage time at a Railway Station, find a minimum number of platforms needed.

Note -

If Train A's departure time is x and Train B's arrival time is x, then we can't accommodate Train B on the same platform as Train A.

Constraints

1 <= N <= 10^5

0 <= a <= 86400

0 < b <= 86400

Number of platforms > 0

Input

The first line contains N denoting the number of trains.

Next N line contains 2 integers, a and b, denoting the arrival time and stoppage time of the train.

Output

A single integer denotes the minimum number of platforms needed to accommodate every train.

Example 1

Input

3

10 2

5 10

13 5

Output

2

Problem 18: Count Pairs

Given an array of integers A, and an integer K find a number of happy elements.

Element X is happy if there exists at least 1 element whose difference is less than K i.e. an element X is happy if there is another element in the range [X-K, X+K] other than X itself.

Constraints

1 <= N <= 10^5

0 <= K <= 10^5

0 <= A[i] <= 10^9

Input

The first line contains two integers N and K where N is the size of the array and K is a number as described above. The second line contains N integers separated by space.

Output

Print a single integer denoting the total number of happy elements.

Example 1

Input

6 3

5 5 7 9 15 2

Output

5

Problem 19 :Critical Planets

The war between the Republic and the Separatists is escalating. The Separatists are on a new offensive. They have started blocking the path between the republic planets (represented by integers) so that these planets surrender due to the shortage of food and supplies. The Jedi council has taken note of the situation and they have assigned Jedi Knight Skywalker and his Padawan Ahsoka to save the critical planets from blockade (Those planets or systems of planets which can be accessed by only one path and may be lost if that path is blocked by separatist).

Skywalker is preparing with the clone army to defend the critical paths. He has assigned Ahsoka to find the critical planets. Help Ahsoka to find the critical planets(C) in ascending order. You only need to specify those planets which have only one path between them and they cannot be accessed by any other alternative path if the only path is compromised.

Constraints

M <= 10000

N <= 7000

Input

The first line contains two space-separated integers M and N, where M denotes the number of paths between planets and N denotes the number of planets.

Next M lines, each contains two space-separated integers, representing the planet numbers that have a path between them.

Output

Clines containing one integer representing the critical planet that they need to save in ascending order of the planet number if no planet is critical then print -1

Time Limit

1

Example 1

Input

3 4

0 1

1 2

2 3

Output

0

1

2

3

Problem 20:Bank Compare

There are two banks – Bank A and Bank B. Their interest rates vary. You have received offers from both banks in terms of the annual rate of interest, tenure, and variations of the rate of interest over the entire tenure. You have to choose the offer which costs you the least interest and reject the other. Do the computation and make a wise choice.

The loan repayment happens at a monthly frequency and Equated Monthly Installment (EMI) is calculated using the formula given below :

EMI = loanAmount * monthly interest rate / (1 – 1 / (1 + monthly interest rate)^(number of years * 12))

Constraints:

1 <= P <= 1000000

1 <=T <= 50

1<= N1 <= 30

1<= N2 <= 30

Input Format:

- First line: P principal (Loan Amount)
- Second line: T Total Tenure (in years).
- Third Line: N1 is the number of slabs of interest rates for a given period by Bank A. The first slab starts from the first year and the second slab starts from the end of the first slab and so on.
- The next N1 line will contain the interest rate and their period.
- After N1 lines we will receive N2 viz. the number of slabs offered by the second bank.
- Next N2 lines are the number of slabs of interest rates for a given period by Bank B. The first slab starts from the first year and the second slab starts from the end of the first slab and so on.

- The period and rate will be delimited by a single white space.

Output Format: Your decision is either Bank A or Bank B.