String Slice (100 Marks)

You need to take string input and two other numbers which will be the start and end point of the slice and you need to print that slice of string.

Input Format

You will be given a function with string and two other integers as arguments.

Constraints

1 <= |S| <= 10^3

Output Format

You need to return the slice of the string.

Sample TestCase 1

Input

Hello Techgig

1

4

Output

ello

/\* Read input from STDIN. Print your output to STDOUT\*/

import java.io.\*;

import java.util.\*;

public class CandidateCode

{

public static void main(String args[] ) throws Exception

{

//Write code here

Scanner sc=new Scanner(System.in);

String string1=sc.nextLine();

String string2="";

int length1=string1.length();

int start=sc.nextInt();

int end=sc.nextInt();

for(int j=start;j<end+1;j++)

{

char c=string1.charAt(j);

string2=string2+c;

}

System.out.println(string2);

}

}

Techgig geek goddess TG3 problem:-

**Chief Election**(100 Marks)

In the city of La La Land, the sports chief is to be elected. Mr. Jani, who himself is a great athlete, is standing for the position. It looks like a cake for him but it is not so. Throughout his career, he has been involved in many controversies and because of this the people are divided. To challenge him, Ms. Ramya has also submitted her application. She is a fierce lady with a great track record but is considered too strict with her discipline norms.

There are multiple departments in the sports authority. On the election day, each department is provided with an integer. It also means multiple persons from the same department have the same integer.

**Election Rules:**

1. Different persons belonging to the same department, have the same integer.
2. Multiple votes with the same integer will be considered as a single unique vote.
3. To win, Mr. Jani should have atleast **M** unique votes. If not, then, Ms. Ramya will be elected as the Chief Officer.

**Note:** If there are 3 people belonging to one department having an integer 5 and they all vote, then it will be considered as a single unique vote.

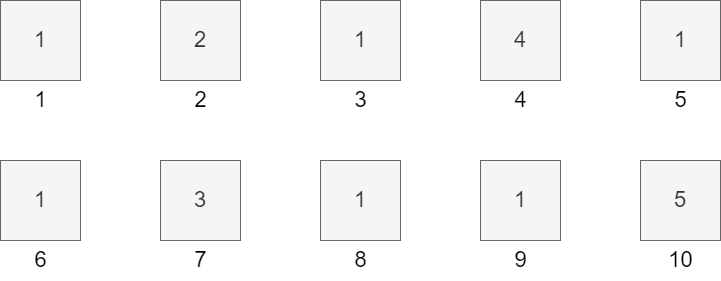
Now, on election day, **N** people have casted their vote. The votes are stored and are presented to you. The authority is asking you to declare the winner as per the rules. Declare the election results and name the new Chief Officer.

**Example:**

Number of people who casted their vote, N = 10

Number of minimum votes required by Mr. Jani, M = 6

The votes casted are given below,



There are only 5 unique votes, 1, 2, 3, 4 and 5. Mr. Jani required minimum 6 unique votes to win so Ms. Ramya is the winner.

**Input Format**

The first line of input consists of number of test cases, T

The first line of each test case consists of two space-separated integers, N and M.

The second line of each test case consists of N space-separated integers.

**Constraints**

1<=T<100

1<= N <=10000

0<= M <=N

1<= Ni <=10000

**Output Format**

For each test case, print **JANI** if Mr. Jani wins otherwise print **RAMYA.** The output is case sensitive.

Sample TestCase 1

Input

2

10 6

1 2 1 4 1 1 3 1 1 5

6 3

1 2 3 4 2 1

Output

RAMYA

JANI

Explanation

**Test Case 1:** As explained in the example.

**Test Case 2:** Minimum unique votes required by Mr. Jani is 3. The number of unique votes casted is 4 (1, 2 3 and 4). The winner is Mr. Jani.

/\* Read input from STDIN. Print your output to STDOUT\*/

import java.io.\*;

import java.util.\*;

public class CandidateCode {

public static void main(String args[] ) throws Exception {

   //Write code here

Scanner sc=new Scanner(System.in);

int T=sc.nextInt();

for(int j=0;j<T;j++)

{

int N=sc.nextInt();

int M=sc.nextInt();

int count1=0;

int count2=0;

int array1[]=new int[N];

for(int i=0;i<N;i++)

{

array1[i]=sc.nextInt();

}

int max=Integer.MIN\_VALUE;

for(int element:array1) //for-each loop

{

if(element>=max)

{

max=element; //max=4; from array1

}

}

int array2[]=new int[max+1];

for(int i=0;i<array2.length;i++) //i<max+1 //same

{

array2[i]=0;

}

for(int el1:array1)

{

array2[el1]++;

}

for(int i=0;i<array2.length;i++) //i<max+1 //same

{

if(array2[i]>1) //checking which elements of array2 are more than 1 because the index of those elements

//are the repeating elements in that array

{

count1=count1+1;

}

}

for(int i=0;i<array2.length;i++) //i<max+1 //same

{

if(array2[i]==1) //checking which elements of array2 are equal to 1 because the index of those elements

//are the non-repeating elements in that array

{

count2=count2+1;

}

}

int result=count1+count2;

if(j>0)

{

if(result>M)

{

System.out.println();

System.out.print("JANI");

}

else

{

System.out.println();

System.out.print("RAMYA");

}

}

else

{

if(result>M)

{

System.out.print("JANI");

}

else

{

System.out.print("RAMYA");

}

}

}

}

}

Atcoder problem:-

**Problem Statement**

You are given a string of length *NN*, *S=S\_1S\_2\dots S\_NS=S1​S2​…SN​*, consisting of 0, 1, and ?.

We like to replace every ? with 0 or 1 so that all of the following conditions are satisfied.

* *SS* contains exactly *KK* occurrences of 1.
* These *KK* occurrences of 1 are consecutive. That is, there is an *i\ (1 \leq i \le N-K+1)i (1≤i≤N−K+1)* such that *S\_i=S\_{i+1}=\dots=S\_{i+K-1}=Si​=Si+1​=⋯=Si+K−1​=* 1.

Determine whether there is exactly one way to replace the characters to satisfy the conditions.

You have *TT* test cases to solve.

**Constraints**

* *1 \leq T \leq 10^51≤T≤105*
* *1 \leq K < N \leq 3 \times 10^51≤K<N≤3×105*
* *SS* is a string of length *NN* consisting of 0, 1, and ?.
* The sum of *NN* across the test cases is at most *3 \times 10^53×105*.

**Input**

The input is given from Standard Input in the following format:

*TT*

*\mathrm{case}\_1*case*1​*

*\vdots⋮*

*\mathrm{case}\_T*case*T​*

Each case is in the following format:

*NN* *KK*

*SS*

**Output**

Print *TT* lines. The *ii*-th line should contain Yes if, for the *ii*-th test case, there is exactly one way to replace the characters to satisfy the conditions, and No otherwise.

**Sample Input 1**Copy

Copy

4

3 2

1??

4 2

?1?0

6 3

011?1?

10 5

00?1???10?

**Sample Output 1**Copy

Copy

Yes

No

No

Yes

For the first test case, turning *SS* into 101, for instance, does not satisfy the conditions since the 1s will not be consecutive. The only way to satisfy the conditions is to turn *SS* into 110.

For the second test case, we may turn *SS* into 1100 or 0110 to satisfy the conditions, so there are two ways to satisfy them.

For the third test case, there is no way to replace the characters to satisfy the conditions.

### Carnival (100 Marks)

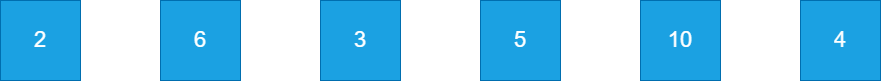
A carnival is taking place in La La Land. There are rides, food joints, and multiple shops for games and shopping. Everyone is enjoying and having a good time. There is one particular game shop which is having the most crowd. Rachel is also there and quite excited about it.

The game shop has a simple game. In the game shop, there are **N** items with prices mentioned on them. You have to divide the items into two groups such that one group has exactly **M** items and the other group has the rest. The division of groups is not random though. While dividing the items into groups, you have to do it in a way such that the difference in the summation of price between the two groups is the maximum.

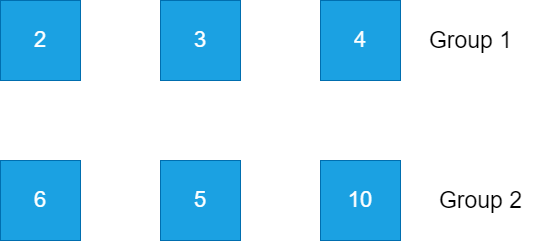
Example:

Number of items, N = 6

Number of items in one group, M = 3



We can divide the items into groups as below,



The summation of Group 1 = 2 + 3 + 4 = 9

The summation of Group 2 = 6+ 5 + 10 = 21

The difference in price = 21 - 9 = 12. Note, this is the maximum difference possible.

Rachel wants to have the gifts but she doesn't know how to divide the items into groups to win. Her friend Mike is very busy with work and so she asks you to help her. Can you determine the maximum price difference between the two groups?

##### Input Format

The first line of input consists of number of test cases, T

The first line of each test case consists of two space-separated integers, N and M.

The second line of each test case consists of N space-separated integers representing the price of N items.

##### Constraints

1<= T <=10

1<= N <=1000

1<= M < N

1<= Price of items <=10000000

##### Output Format

For each test case, print the maximum price difference possible.

###### Sample TestCase 1

Input

2

6 3

2 6 3 5 10 4

5 2

1 3 2 2 3

Output

12

5

###### Explanation

**Test Case 1:** As explained in the example.

**Test Case 2:**

Number of items = 5

Number of items in one group has to be 2.

The other group will have 3 items.

**Group 1 = {1, 2}**

**Group 2 = {2, 3, 3}**

**Summation of Group 1 = 1+ 2 = 3**

**Summation of Group 2 = 2 + 3 + 3 = 8**

**Maximum price difference possible = 8 - 3 = 5**