Problem 1:

Nisha is playing a game with his friend Misha. They have a pile containg N coins. Players take alternate turns, removing some coins from the pile. On each turn, a player can remove either one coin or coins equal to some prime power (i.e. pxcoins, where p - prime number and x - positive integer). Game ends when the pile becomes empty. The player who can not make a move in his turn loses.

Nisha plays first. Your task is to find out who will win the game, provided that both of the player play optimally.

Input format: Input contains a single integer n.

Output format: Output consists of a one word - the name of the winner of the game. Print 'NIsha' if Nisha wins the game, print 'Misha' otherwise.

Constraints $1 \leq N \leq 109$

Sample Input 1: 1

Sample Output 1: NIsha

Sample Input 2: 8

Sample Output 2: NIsha

Sample Input 3: 0

Sample Output 3: Invalid Input

Explanation:

Example case 1. NIsha will remove the only coin from the pile and will win the game.

Example case 2. NIsha will remove all 8 coins from the pile and win the game. Nisha can remove 8 coins because 8 is a prime power, as 8 = 23.

Problem 2

Tarun and Varun are learning game theory. Today they came up with a new game.

The rules are as follows:

1. Initially they have a string consisting of lowercase english alphabets only.

2. Each of them take turn.

3. In each turn, a player removes some alphabets from the string (at least one). But the condition is all the alphabets removed must be same (see explanation of the test cases).

4. The player who is unable to make a move loses.

You are given the string and the player who makes the first move. You have to print the name of the winner.

Input format:

Input will contain two strings separated by a space. First is the name of the player who starts the game (“Tarun” or “Varun”), second is the input string with which the game is to be played.

Output format: Output consists of a word. Print “Tarun”, if Tarun wins, else print “Varun”.

Constraints : Input string will contain lower case alphabets only.

Length of input string will not exceed 100.

Sample Input 1: Tarun aaa

Sample Output 1: Tarun

Sample Input 2: Varun ababa

Sample Output 2: Varun

Sample Input 3: Tarun abacd

Sample Output 3: Tarun

Explanation

1. Tarun can remove all three 'a' and win.

2. Varun can remove a single 'a' in his turn. Tarun will be left with two 'b' and two 'a'. If he removes both 'b' Varun can remove both 'a' and win, else if he removes a single 'a', Varun will be left with 'a', 'b' and 'b'. Varun can remove a single 'b', then Tarun will be left with 'a' and 'b'. Whatever he removes, Varun will remove the last letter and win.

Note: Remember neither of them can remove letters like {'a','a','b'}, {'a','b','a','b'} etc as the condition says all the letters removed must be same.

Problem 3

Given two integers X and Y and an array of N integers. Player A can decrease any element of the array by X and Player B can increase any element of the array by Y. The task is to count the number of elements that A can reduce to 0 or less. They both play optimally for infinite time with A making the first move.

Note: A number once reduced to zero or less cannot be increased.

Examples: Input: a[] = {1, 2, 4, 2, 3}, X = 3, Y = 3

Output: 2

One of the way:

A reduces 2 to -1

B increases 1 to 4

A reduces 2 to -1

B increases 3 to 6 and the game goes on.

Input: a[] = {1, 2, 4, 2, 3}, X = 3, Y = 2

Output: 5

Explanation:

Since the game goes on for infinite time, we print N if X > Y.

Now we need to solve for X ≤ Y. The numbers can be of two types:

Type 1: Those which do not exceed X on adding Y say count1 which can be reduced to ≤ 0 by A.

Type 2:Those which are < X and exceed X on adding Y say count2 only half of which can be reduced to ≤ 0 by A as they are playing optimally and B will try to increase any one of those number so that it becomes > X in each one of his turns.

So, the answer will be **count1 + ((count2 + 1) / 2).**

Input format:

First line of input consists of an integer n, corresponds to the size of array.

The next n lines of input corresponds to the elements in an array.

The next lines consists of an integer number X

The next lines consists of an integer number Y

Output format:

Output consists of an integer which corresponds to the count of elements 0 or less.

Sample Input :

5 (size of the array)

1

2

3

4

5

3 (X)

3 (y)

Sample Output : 2

Problem4

You are given an array A[] of n-elements. There are two players Alice and Bob. A Player can choose any of element from array and remove it. If the bitwise XOR of all remaining elements equals 0 after removal of selected element, then that player looses.

Note : Each players play game alternately. Find out winner if both of the players play optimally. Alice starts the game first. In case only one-element in array consider its value as the XOR of array.

Examples: Input: A [] = {3, 3, 2}

Output: Winner = Bob

Explanation:

Alice can select 2 and remove it that make XOR of array equals to zero

Also if Alice choose 3 to remove than Bob can choose any of 2 or 3 and finally Alice have to make his steps.

Input: A[] = {3, 3}

Output: Winner = Alice

Explanation: As XOR of array is already zero Alice can't select any element to remove and hence Alice is winner.

Explanation:

Let's start the solution step by step. We have total of three option for the XOR of array and this game.

XOR of array is already 0: In this case Alice will unable to make a move and hence Alice is winner.

XOR of array is not zero: Now, in this case we have two options, either size of array will be odd or even.

CASE A: If the array size is odd then for sure Bob will win the game.

CASE B: If the array size is even then Alice will win the game.

Above conclusion can be proved with the help of mathematical induction.

Sample Input 1:
3 (Size of array)
3
3
2
Sample Output 1: Bob
Sample Input 2:
2 (Size of array)
3
3
Sample Output 2: Alice

Problem 5

Given an array arr[] of integers, two players A and B are playing a game where A can remove any number of non-zero elements from the array that are multiples of 3. Similarly, B can remove multiples of 5. The player who can't remove any element loses the game. The task is to find the winner of the game if A starts first and both play optimally.

Examples: Input: arr[] = {1, 2, 3, 5, 6}

Output: A

3 and 6 are the elements that A can remove.

5 is the only element that B can remove.

A can remove 3 in his first move then B will have to remove 5. In the next turn, A will remove 6 and B will be left with no more moves to make.

Input: arr[] = {3, 5, 15, 20, 6, 9}

Output: A

Explanation:

Store the count of elements only divisible by 3 in movesA, count of elements only divisible by 5 in movesB and the elements divisible by both in movesBoth. Now,

If movesBoth = 0 then both the players can remove only the elements which are divisible by their respective number and A will win the game only when movesA > movesB.

If movesBoth > 0 then in order to play optimally, A will remove all the elements that are divisible by both 3 and 5 so that B is left with no elements to remove from the common elements then A will be the winner only if movesA + 1 > movesB

Sample Input 1:
6 (Size of array)
3
5
15
20
6
9
Sample Output 1: A
Sample Input 2:
3 (Size of array)
1
3
5
Sample Output 2: B

Problem 6

There are two players P1 and P2 and two piles of coins consisting of M and N coins respectively. At each turn, a player can choose only one of the piles out of these and discard the other one. This discarded pile cannot be used further in the game. The choosed pile is further divided by the player into two piles of non zero parts. The player who cannot divide the pile i.e. the number of coins in the pile is < 2, loses the game. The task is to determine which player wins if P1 starts the game and both the players play optimally.

Examples: Input: M = 4, N = 4

Output: Player 1

Player 1 can choose any one of the piles as both contain the same number of coins and then splits the chosen one (the one which is not chosen is discarded) into two piles with 1 and 3 coins respectively.

Player 2 can choose 3 coins pile and then splits the chosen one into two piles with 1 and 2 coins respectively.

Player 1 can choose 2 coins pile and then splits the chosen one into two piles with 1 coin each.

Now, player 2 is left with no move (as both the remaining piles contain a single coin each which cannot be split into two groups of non-zero coins).

Input: M = 1, N = 1

Output: Player 2

There's no move to make.
Simply check if any of the pile consists of even number of coins. If yes then Player 1 wins else Player 2 wins.
Sample Input 1:
4 (M)
4 (N)
Sample Output 1: P1
Sample Input 2:
1
1
Sample Output 2: P2