QUESTION 1

Maximum Difference in an Array

QUESTION DESCRIPTION

The maximum difference for a pair of elements in some array a is defined as the largest difference between any a[i] and a[j] where i < j and a[i] < a[j].

The declaration for a function named maxDifference (which takes array a as a parameter) is provided for you in the editor. Complete the function so that it calculates and returns the maximum difference for a; if no such number exists (e.g.: if a is in descending order and all a[j] < a[i]), return -1.

Input Format

The task of reading input (defined below) from stdin, assembling it into array a, and and calling maxDifference(a) is already handled for you by the locked code in the editor.

The first line contains N (the number of elements in array a).

The N subsequent lines each contain a single element of a; the i^{th} line of input (where 0 < i < N-1) contains element a[i].

Constraints

$1 \leq N \leq 10^6$

$-10^6 \leq a[i] \leq 10^6 \; \forall \; i \in [0, N-1]$

Output Format

Your maxDifference function should return the maximum difference in a. Printing to stdout is handled for you by the locked code in the editor.

Sample Input 0

```
7
2
3
10
2
4
8
1
```

Sample Output 0

```
8
```

Sample Input 1

```
6
7
9
5
6
3
2
```

Sample Output 1

```
2
```

Explanation

Sample Case 0: n = 7, a = {2, 3, 10, 2, 4, 8, 1}

As a[2] = 10 is largest element in the array, we must find the smallest a[i] where $0 \leq i < 2$. This ends up being 2 at index 0.

We then calculate the difference between the two elements: a[2] − a[0] = 10 − 2 = 8, and return the result (8).

Note: While the largest difference between any two numbers in this array is 9 (between a[2] = 10 and a[6] = 1), this cannot be our maximum difference because the element having the smaller value (a[6]) must be of a lower index than the element having the higher value (a[2]). As 2 is not less than 6, these elements cannot be used to calculate our maximum difference.

Sample Case 1: n = 6, a = {7, 9, 5, 6, 3, 2}

The maximum difference returned by our function is a[1] − a[0] = 9 − 7 = 2, because 2 is the largest difference between any a[i] and a[j] satisfying the conditions that a[i] < a[j] and i < j.

QUESTION 2 Permutations divisible by 8

QUESTION DESCRIPTION

You are given an integer N. Is there a permutation of that integer's digits that yields an integer divisible by 8? For example, if the number N = 123, then {123, 132, 213,
231, 312, 321} are the possible permutations and 312 is divisible by 8.

Constraints
$1 \leq T \leq 45$
$0 \leq N \leq 10^{110}$

Input Format
The first line contains an integer **T** that gives the number of test cases.
T lines follow, each containing one integer **N**.

Output Format
For each test case, print YES if there exists at least one way of re-arranging its digits such that it is divisible by 8, and print NO otherwise.

Sample Input #00

```
2
61
75
```

Sample Output #00

```
YE
S
N
O
```

Explanation #00
Test case #1: 16 is permutation of 61 which is divisible by 8.
Test case #2: None of permutation of 75, {57, 75}, are divisible by 8.

QUESTION 3

4th Bit

QUESTION DESCRIPTION

Complete the fourthBit function in your editor. It has 1 parameter: an integer, num. It must return the binary integer (i.e.: 0 or 1) corresponding to the 4^{th} least-significant bit of the 32-bit value passed to it as an argument.

Input Format

The locked stub code in your editor reads a single integer, num, from stdin and passes it to your function.

Constraints

num is a 32-bit integer.

Output Format

Your function must return the binary integer corresponding to the 4^{th} least-significant bit of the integer argument passed as num. This is printed to stdout by the locked
stub code in your editor.

Sample Input 0

32

Sample Output 0

0

Sample Input 1

77

Sample Output 1

1

Explanation

Sample Case 0:
The integer $(32)_{10}$ converts to $(100000)_2$. If we 1-index each bit from least to most significant, they are indexed as 654321. Because the bit at index 4 is 0, we return 0 as our answer.

Sample Case 1:
The integer $(77)_{10}$ converts to $(1001101)_2$. If we 1-index each bit from least to most significant, they are indexed as 7654321. Because the bit at index 4 is 1, we return 1 as our answer.