

## Assignment 4

### Instruction:

1. Attempt any 4 in each section .

### SectionA

#### P1

Penny likes sorting. But not today. Today, she will multiply numbers. Penny has some files, numbered from 0 to  $n-1$ . Each file has an Alien Registration Number (ARN). She took out the file of an alien and found out that the file was encrypted. What does that mean? Why is the file encrypted? No idea. Let's just open the file. The key to opening the  $n^{\text{th}}$  file is the product of all ARNs except that of the  $n^{\text{th}}$  file. For each given file number, tell the key to opening the file.

#### Input:

First line contains number of test cases  $T$   
For each test case First line contains  $n$ , the number of files. Next line contains  $n$  integers,  $ARN_0$  to  $ARN_{(n-1)}$ ,  $ARN_i$  determining the ARN of the alien in file number  $i$   
Next line contains  $q$ , the number of files to be opened  
Next line contains  $q$  space separated integers  $q_i$

#### Output:

For each query  $q_i$ , print the key to opening the file number  $q_i$  followed by a space  
Answer each test case in a separate line

#### Constraints:

Product of all keys  $\leq 1000\ 000\ 000\ 000\ 000\ 000$   
 $1 \leq n \leq 1\ 000\ 000$   
 $0 \leq \text{Each ARN} \leq 1,00,00,00,000$   
 $1 \leq q \leq 1\ 000\ 000$   
 $0 \leq q_i < n$

#### Sample Input

```
2
4
2 4 3 8
2
2 3
5
0 1 5 6 4
3
0 2 3
```

**Sample Output:**

```
64 24
120 0 0
```

**P2.**

## Continuos Xor

Penny has an array of non-negative numbers. Find the segment of consecutive elements, such that the xor of all numbers from this segment is maximum possible. Just print the maximal XOR value.

**Input:**

Each line has number of test cases, T  
Each test case has n followed by n space separated integers

**Output:**

One integer, the XOR required

Example:

Input:

```
3
5 1 2 1 1 2
3 1 2 7
4 4 2 4 8
```

Output:

```
3
7
14
```

**P3.**

## Chess

Penny and Leonard are playing chess. But Penny likes sorting, so she tried to sort the cells on the chessboard. Consequently, the chessboard was damaged. Leonard replaced few blocks in this board, and now the chess board may not be proper. A proper chessboard doesn't have any adjacent cells with the same color and the upper left cell is white. As you would expect, it is an 8X8 chessboard. Given, for each cell, is a character 'W' or a character 'B', which tells if it is black or white. Now, you have to decide if you can fix the chessboard using just magicRotate. You can use magicRotate as many times as you like. In MagicRotate, you can choose any row of the board and cyclically shift the cells of the chosen row, that is, put the last (rightmost) square on the first place in the row and shift the others one position to the right.

For example, if the first line of the board is "BBBBBBWW", then after one magicRotate, it will look like that

"WBBBBBBW".

**Input:**

First line is number of test cases

In each case,

The input consists of exactly eight lines. Each line contains exactly eight characters "W" or "B" without any spaces: the j-th character in the i-th line stands for the color of the j-th cell of the i-th row of the elephants' board. Character "W" stands for the white color, character "B" stands for the black color.

Consider the rows of the board numbered from 1 to 8 from top to bottom, and the columns from 1 to 8 from left to right. The given board can initially be a proper chessboard.

Test cases are separated by a blank line

**Output:**

In a single line print "YES" (without the quotes), if we can make the board a proper chessboard and "NO" (without the quotes) otherwise.

**Example:**

2

WBWBWBWB  
BWBWBWBW  
BWBWBWBW  
BWBWBWBW  
WBWBWBWB  
WBWBWBWB  
BWBWBWBW  
WBWBWBWB

WBWBWBWB  
WBWBWBWB  
BBWBWWWB  
BWBWBWBW  
BWBWBWBW  
BWBWBWWW  
BWBWBWBW  
BWBWBWBW

**Output:**

YES

NO

**Explanation:**

In the first sample you should shift the following lines one position to the right: the 3-rd, the 6-th, the 7-th and the 8-th.

In the second sample there is no way you can achieve the goal.

**P4.**

Given 2 unsigned integers N and P . You need to print the number formed by rotating the bits of N, P times/Units from left to right. For example if N and P are 2 and 2. Then the bit pattern of 2 is 000010 . after rotating by 2 units we get 001000. Try doing without finding the bits. (Using bitwise operators)

**Input:**

T - Number of test cases  
each test case contains the following  
N and P

**Output:**

1 line per test case containing the number formed after rotating the bits.

**Example:****Input:**

2  
2 2  
2 30

**Output:**

8  
2147483648

**Constraints:**

$0 < T < 10^5$   
 $0 < N < 2^{64}$   
 $0 < P < 64$

**P5:**

Appu is trying to learn chess .He took a chessboard , placed all the knights he had in his hand on the chessboard . Please help Appu find out whether any 2 knights attack each other or not.

**Input:**

n - size of the chess board  
n x n array with 0 for empty positions and 1 for knight position.

**Output:**

YES if there is an attack  
NO else

**constraints:**

$0 < n < 100$

**Example:**

4  
1 0 0 0  
0 1 0 0  
0 0 1 0  
0 0 0 1

**Output:**

NO

**P6:**

Appu needs to find out whether the given number is a power of 2 or not . He takes your help.

**Input:**

T - test cases  
T lines containing 1 number each.

**Output:**

1 line for each input  
YES if number is a power of 2  
NO if not

**Note:** Try solving using the bit wise operators.

**Example****Input:**

3  
2  
1  
5

**Output:**

YES  
YES  
NO

**Constraints**

$0 < T < 3 \cdot 10^6$   
 $0 < N < 10^8$

## SectionB

P7.

### Binary Search

Penny is a MAD Scientist. She completed her B.tech+MS in five years, did her PhD in three years, and finally joined Ministry for Alien Defense. So, she is now a Ministry for Alien Defence Scientist (MAD Scientist).

During her work at MAD, she often felt the need to take out the files for a particular alien and Search through all the documents for activity history. The files were sorted according to the Alien Registration Number (ARN). Because Penny likes sorting.

But, she feels it is too cumbersome to look through the files every time. You need to help her, so that MAD can function properly.

### Input:

First line contains  $n$ , the number of files.

Next line contains  $n$  integers, determining the ARN of the alien in the file, from file 0 to file  $n-1$ . Each  $ARN \leq 1,00,00,00,000$ . The next line contains  $q$ , the number of queries.

Next  $q$  lines contain a number( $x$ ) each.

### Output:

Answer each query in one line

For each query  $x$ , if an alien with  $ARN=x$  exists in the files, print the file number  
else print -1

### Constraints:

$1 \leq n \leq 10,00,000$

$0 \leq q \leq 1,00,000$

### Example:

```
10
2 3 45 455 456 458 2345 9123 23414 100002
5
2
45
23414
8765
3
```

### Output:

```
0
```

2  
8  
-1  
1

**P8.**

**Recursive XOR**

Penny has an array. She wants to select 0 or more elements from this array, such that their XOR is maximum. Find this maximum value.

**Input:**

first line is T  
followed by T testcases  
each Test case has n followed by n numbers

**Output:**

Give the required number for each case in separate lines

**Constraints:**

0<=each number <=1000000000 (9 zeroes)  
1<=n<=15  
1<=T<=100

**Example:**

3  
3 1 2 3  
1 8  
8 561109929 434745088 231000930 277665073 88537520 657009779  
814498587 425520023

**Output:**

3  
8  
1068617456

**P9.**

**Character Matrix**

Penny has a \*square\* matrix of characters. It consists of '.' or 'o'. She wants to find out if there is a straight line of 5 'o's in a row (Diagonal, horizontal or vertical)

**Input:**

First line is number of test cases  
In each case,  
First line has n and m -> number of rows and columns  
Each of next n lines has a string of length m

**Output:**

```
print "found" if there is a straight sequence of 5
'o's in a row else print "not found"
```

**Constraints:**

```
1<=n,m<=100
```

**Example:**

```
4
```

```
6 6
```

```
....O.
```

```
...O..
```

```
..O...
```

```
.O....
```

```
O.....
```

```
.....
```

```
5 5
```

```
OOOO.
```

```
.OOOO
```

```
OOOO.
```

```
.OOOO
```

```
.....
```

```
5 5
```

```
OOOO.
```

```
.OOOO
```

```
OOOO.
```

```
.OOOO
```

```
.....O
```

```
6 6
```

```
O.....
```

```
.O.....
```

```
..O....
```

```
...O..
```

```
....O.
```

```
.....
```

```
5 5
```

```
O....
```

```
O.O..
```

```
O.O.O
```

```
O.O..
```

```
O....
```

**Output:**

```
found
```



not found  
found  
found  
found

**P10.**

Appu loves ice-cream .He buys many ice cream in a fair .He decided to eat them later and plans to store the ice cream in a fridges.Each fridge can accomodate atmost K ice creams . There fore he decides to divide the number of ice creams into 2 parts(parts have equal number of icecreams if the number of ice-creams is even),(one part has one more ice cream than the other if the number of ice-creams in odd).He furthur divides each part into 2 halves untill the number of ice-creams fit in the fridge. As usual we need to help Appu to decide how many fridges he needs to buy if he stores all the icecreams in the fridges.

**Input:**

T - number of testcases  
N K - N-number of ice-creams K - maximum number of icecream in each fridge  
Ouput:  
L - number of fridges required for each test case

**Example:**

**INPUT:**

2  
14 3  
15 1

**Output:**

6  
15

**Constraints:**

$0 < T < 10^5$   
 $0 < N < 1000$   $0 < K < 100$

**P11.**

Appu decides to buy R different chocolates . He goes to a shop . the shop contains many types of chocolates(N). As usual Appu is poor at maths . Help him to find the number of ways in which we can select R chocolates from N distinct types of chocolate.

Note: Try to use  $nCr = nCr-1 + (n-1)C(r-1)$  along with memoization(look at the number of queries!!)

As ncr is very big for large n you are required to calculate  $\text{ncr} \% 1000000007$ .

**INPUT:**

T - test case  
T following lines contain  
N R

**Output:**

A line per testcase giving the answer.

**Constraints:**

$0 < T < 10^6$   
 $1 < N < 1000$   
 $0 < R < N$

**Input:**

3  
5 2  
35 12  
100 50

**Output:**

10  
834451800  
538992043

**P12.**

Appu owns a shop . His manager presented him with a report of his profit for the past n years. He wants to check this profits for few continuous periods of years . Suppose he has a profits of p1 p2 p3 p4 in 1st 2nd 3rd and 4th years then the he wants the queries like what is his profits from 2nd to 4th year (both included) answered.)

**Input:**

N - the number of years for which the report is provided.  
next line contains the N numbers which are the profits during each year .  
following are the queries where he gets 2 numbers A and B in each line  
-1 -1 (End of queries is denoted by -1 and -1)

**Output:**

1 line for each output giving the total profit from A to B year(Both included).

**Example:****Input:**

```
5
1 2 3 4 5
1 1
1 5
2 4
-1 -1
```

**Ouput:**

```
1
15
9
```

**Constraints:**

```
0 < N < 10^5
number of queries will be 1000
0 < pi < 10^9
```

**SectionC****P13.****Exponent**

Penny has to find  $a$  to the power  $b$ . She multiplies  $a*a*a...$   $b$  times. You have to do the same task faster.

In each test case, given a positive number  $a$  and a positive exponent  $b$ , calculate  $a$  to the power  $b$  modulo 1000000007.

Denote  $a$  power  $b$  by  $a**b$ .

Calculate  $(a^{b \% 1000000007})$

**Input:**

First line is number of test cases  $T$   
Next  $T$  lines contain two numbers  $a$  and  $b$

**Output:**

In each line, print the value of  $(a^{b \% 1000000007})$

**Example:**

```
3
2 3
342 5423
3434252 52354423
```

**Output:**

```
8
743935427
```

185274193

**Constraints:**

1<=T<=1000  
a<=1000000000 (9 zeroes)  
b<=1000000000 (9 zeroes)  
(a<sup>b</sup>) may be very very large

**Hint:**

Sheldon says answer is less than 1000000007 (Why?)  
a,b, answer,1000000007 all fit inside int datatype  
Product of two ints will always fit inside long long int  
datatype (Why?)  
(a\*b)%M = ((a%M)\*(b%M))%M  
If you take x%1000000007 each time, entire product will  
always be in limit, even though a\*\*b may be very large  
**Hint to recursive step:** If you know a\*\*(b/2), can you find  
out a\*\*b ?

**Scoring:**

30 points for small test cases  
100 points for large test cases (like in the example above)

**P14**

**Merge Sort**

Penny likes sorting. Whenever she is free, she sorts numbers. She sorts numbers in the morning, to sharpen her thinking. Amy thinks that this sharpening is not particularly effective, but it doesn't matter. Because Penny likes sorting.

Now, we need to sharpen the mind of our computers. So, we need to teach them how to sort. Given n numbers, print them in sorted order.

**Input**

First line contains T, the number of test cases  
In each test case,  
First line contains n, the number of numbers to sort  
Next line contains n space separated numbers a[1],  
a[2], .. a[n]

**Output**

For each test case, print the sorted list of given numbers a[i] 1<=i<=n . Print a space after \*each\* number.  
Because Penny likes sorting.

**Constraints**

1<=T<=30

1<=n<=100000  
-1000000000000000000<=a[i]<=1000000000000000000 (that  
is 18 zeroes, btw)

**Example**

2  
3  
4 2 1  
5  
0 2 3 5 2

**Output:**

1 2 4  
0 2 2 3 5

**P15**

**Suffix Array**

Penny wants to learn about Suffix arrays and is excited  
about its uses. Luckily, she will learn it herself,  
and you don't need to use it here. Phew!

You are given 2 strings, A and B

Among all possible substrings of A and B, print the k'th  
lexicographically smallest substring

k>=1

If there are n occurrences of a substring, consider it n  
times

example: for 'aaaa', 'a' as a substring will be considered  
4 times, 'aa' will be considered three times

**Constraints:**

Length of A and B <=6

**Input:**

First line contains number of test cases

For each case,

Next two lines contains two strings A and B

Next line contains q, the number of queries

Next line contains q space separated integers, for each  
query

**Example:**

2  
ace  
bdf  
12  
1  
2  
3

4  
5  
6  
7  
8  
9  
10  
11  
12  
UHEF  
EYR  
2  
3  
14

**Output:**

a  
ac  
ace  
b  
bd  
bdf  
c  
ce  
d  
df  
e  
f  
EF  
UHEF

**Explanation:**

In first test case, substrings of a are a, ace, ac, ce, e,.... and those of bdf are b,bd,df,bdf,....

Sorting all the substrings, list is ['a', 'ac', 'ace', 'b', 'bd', 'bdf', 'c', 'ce', 'd', 'df', 'e', 'f']

For a query x, the xth element is the answer.

**P16**

Given a sequence of numbers find the lexicographically next permutations of that sequence .If there does exit such a sequence then print the given permutation itself.

the lexicographical permutation sequence of 1 2 3 is

1 2 3  
1 3 2  
2 1 3  
2 3 1  
3 1 2

3 2 1

note that 3 2 1 has no number which is next lexicographical permutation.

**Input:**

T - testcases  
each testcase contains the following  
n - Number of numbers in the sequence  
a1 a2 a3 .... an - a line of n numbers

**Output:**

a single line for each test case printing the next lexicographic permutation

**Example**

**Input:**

4  
3  
1 2 3  
3  
3 2 1  
7  
1 2 4 3 6 7 5  
4  
3 4 2 1

**Output:**

1 3 2  
3 2 1  
1 2 4 3 7 5 6  
4 1 2 3

**Constraints:**

$0 < T < 100$   
 $0 < n < 1000$   
 $-10^9 < a_i < 10^9$

**P17**

Appu is now fed up with maths learning maths directly and tries its hand in using cooking to learn maths. Appu has N litres of flour and he knows for each digit what is the amount of flour required to make roti in the shape of that digit. Appu doesn't like making 0's. He can make many digits using the flour . Help him to make the maximum number out of the flour.If no number can be made print -1.

**Input**

N number of litres of flour present.  
next line contains 10 number 1 each for each digit except .  
Output:  
Maximum number which can be made out of the flour.

**Example:**

**Input:**

5  
5 4 3 2 1 2 3 4 5

**Output:**

55555

**Constraints:**

$0 < N < 10^6$   
 $0 < a_i < 10^5$

**P18:**

Appu opens a new bank . In order to attract customers ,  
Appu comes with a scheme . The scheme is as follows . He  
collects N rupees and gives back  $N/2 + N/3 + N/4$  rupees .  
chottu on knowing this wants to maximize his money . Please  
do help him.

**Input:**

12  
2

**Output:**

13  
2

Note: You need to scan till end of the file.

**Constraints:**

There will be no more than 10 numbers in the file  
and each number is from  $0 < n < 10^8$