**A Remember This Day?**

Everyday is special. But the different days in February are mostly special as these are celebrated throughout the whole world. Among all this special days, one of the day is very special to the people of Bangladesh. It is called International Mother Language Day. We are not going to say what day it is.

**Input**

A date from the calendar of the month of February 2021.

**Output**

You have to print “International Mother Language Day!” if the given date represents it. Otherwise print “I don’t care!“

|  |  |
| --- | --- |
| **Sample input** | **Sample output** |
| 7 February 2021 | I don’t care! |
| 14 February 2021 | I don’t care! |

# **B Duplicate Removal**

Write a program that will take (**m x n**) matrix of integers as input. You have to replace all the duplicate integers in the matrix by **-1** except the first occurrence when reading as row-major order.

## **Input**

First line consists of two integers m, n (**0< m,n <= 1000**). Next **m** lines each consists of **n** space separated integers. All the integers will be in range **[1,100000]**

## **Output**

Updated matrix. Elements should be space separated. Each row will be in a new line.

|  |  |
| --- | --- |
| **Sample input** | **Sample output** |
| 3 3  1 7 3  7 4 5  3 5 6 | 1 7 3  -1 4 5  -1 -1 6 |
| 2 6  2 2 2 2 2 2  6 5 4 3 2 1 | 2 -1 -1 -1 -1 -1  6 5 4 3 -1 1 |

**C Game of Odd-Even**

This game starts with a binary string **bstr** and an integer **x**. In each step,

1. If **x** is 0, then the **bstr** is kept as it is.
2. Otherwise,
   1. If **x** is odd number, then all the 1’s in the corresponding **bstr** is replaced by **bstr** with integer **(x-1)**. All the zero’s are kept as it is.
   2. If **x** is even number, then all the 0’s in the corresponding **bstr** is replaced by **bstr** with integer **(x-1)**. All the one’s are kept as it is.

For better understanding, let **bstr** = “10” and **x** = 3. Then, the steps of the game are as follows:

* (10, 3)
* (10, 2)0
* 1(10, 1)0
* 1(10, 0)00
* 11000

**Input**

A line with a binary string **bstr** and an integer **x**.

**Output**

Final string at the end of the game in a line.

**Conditions:**

1 <= Length of **bstr** <= 4

1 <= **x** <= 20

|  |  |
| --- | --- |
| **Sample input** | **Sample output** |
| 10 3 | 11000 |
| 101 4 | 111010101101101010111 |

**D Word Separation**

You will be given a line of input. You have to print the available words in the line in lexicographic order. A word is defined as the sequence of consecutive alphabets. For example, abc, GtA, assasins, etc are words and 1ad2, GTA5, 3tc are not considered as words. But, ad is considered as word by trimming 1 and 2 from 1ad2.

**Input**

A single line of characters. Input consists of any possible printable characters.

**Output**

All the words from the input line as the definition in lexicographic order.

**Conditions:**

There would be at most 100 words in the input line and each word will be a length of at most 100.

|  |  |
| --- | --- |
| **Sample input** | **Sample output** |
| This is a word problem. | This  a  is  problem  word |
| I will get full marks in this problem. | I  full  get  in  marks  problem  this  will |

**E Student Database**

In this problem, you will be given the information of some students. You have to arrange the information

**Input**

First line consists of an integer **N** indicating the number of students.

There will be **N** blocks of information for N students. Each block will be as:

1. A line with an integer **id** representing the unique identity of a student.
2. A line consisting the **name** of the student. Name consists of alphabets and spaces only.
3. A line with five integers representing the final marks of the student for the subjects **Subject1, Subject2, Subject3, Subject4, Subject5** respectively.

**Output**

You have to arrange each student in a line following the sequence.

* **Place** of the student, right justified in 5 characters. Place should be calculated by the total marks from the 5 subjects. Place will start from 1 and the student with maximum marks will get lower place number. If two students get equal total marks, then the student with lower id number will get lower place number.
* **id** should be arranged after Place, right justified in 5 characters.
* A single space
* **Name** of the student should be arranged after id, left justified in 20 characters.
* **Total marks** should be arranged after the name of the student, right justified in 5 characters.
* After the total marks, marks **Subject1, Subject2, Subject3, Subject4, Subject5** should be arranged respectively. Each marks should right justified in 5 characters.

Arrangement should be printed in ascending order of the place number.

**Conditions**

**1 <= N <= 1000**

**1 <= id <= 10000**

**1 <= size of name <= 20**

**0 <= marks of each subject <= 100**

|  |
| --- |
| **Sample Input** |
| 3  1254  Steve Rogers  79 85 92 49 63  5421  Bruce Wayne  89 57 67 95 71  1234  Mr or Mrs BCS  100 100 100 100 100 |
| **Sample Output** |
| 1 1234 Mr or Mrs BCS 500 100 100 100 100 100  2 5421 Bruce Wayne 379 89 57 67 95 71  3 1254 Steve Rogers 368 79 85 92 49 63 |

**F Sum of Primes**

In this problem, you will be given two positive integer **M** and **N**. You have to print the sum of all the prime numbers in the range [**M, N**].

**Input**

First line is an integer **T** which indicates the number of test cases.

Each of the next **T** line consist two integers **M** and **N** which is the range defined in the description.

**Output**

For each line of input, print the sum on integers in [**M, N**] in a separate line.

**Conditions**

**1 <= T <= 100000**

**1 <= M <= N <= 100000**

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 3  1 5  6 10  11 15 | 10  7  24 |
| 5  1 10  2 5  5 9  10 30  7 9 | 17  10  12  112  7 |