**Instructions:**

* *Mobile, Pen drive and any types of electrical devices is prohibited to keep aside.*
* *Any kind of printed materials except while papers is prohibited to keep aside.*
* *Staying on the allocated seat is must. In case, there is any technical issues, ask the responsible instructors for any changes of seat.*
* *You must log into google classroom to get and submit the assignment. Any other means will not be allowed in general.*
* *Talking, helping others and any types of idea sharing will be considered as an illegal activities in evaluation time.*
* *In this evaluation, you are allowed to use operators, conditionals, loop. Only stdio.h and math.h can be used.*
* *Marks are distributed as: 60% for case passing, 20% for code correction and 20% for code structure.*

**--------------------------------------------------- A ----------------------------------------------------------**

Given two integer m and n. You need to print all the odd numbers in the range [m, n].

**Input**

Input consists a line with two integers m and n. 0 <= m, n <= 1000000.

**Output**

Print the result in a single line. Print “empty” if there is no number to print. See the sample

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 1 10 | 1 3 5 7 9 |
| 2 2 | empty |

**--------------------------------------------------- B -----------------------------------------------------------**

Given integers A and B, print the value of GCD(A, B). GCD is greatest common divisor.

**Input:**

Input will consist of two integers, A and B (-1018 <= A, B <= 1018) in one line. There will never be two zeroes.

**Output:**

Output will consist of 1 line: the value of GCD.

**Constraint:**

*You are not allowed to use recursive function and nested loop.*

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 12 18 | 6 |
| 6 3 | 3 |
| 13 41 | 1 |

**--------------------------------------------------- C --------------------------------------------------**

We know that C works for ASCII characters. Every ASCII character is mapped into a ASCII value whether it is an alphabet or a digit or a special character.

**Input:**

A series of valid ASCII characters. It is guaranteed that the total number of input character will not exceed 10000000.

**Output:**

After taking all the characters as input you should print some statistics. The statistics contains the number of vowels, consonants and digits. See the sample I/O for clarification.

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| This is a character variable problem. I am going to take 120 characters as input and to print some statistics. | Input Completed:  Total Vowels: 33  Total Consonants: 53  Total Digits: 3 |
| This is again a character variable problem. This time, I am going to take not more than 1000 characters as input. | Input Completed:  Total Vowels: 35  Total Consonants: 51  Total Digits: 4 |

**--------------------------------------------------- D --------------------------------------------------**

We are interested to find the result for the following expression.

In this problem, you will be asked to find the result of the given expression up to nth terms. Moreover, an another issue to solve this problem is that the result could be a very large number restricting us to store in any C/C++ variables. Thus, you need to print the result after doing a modulo operation with 1000000007.

**Input:**

You will be given a single integer n. 1 <= n <= 100000.

**Output:**

Print the result value in a line as it is asked for. See the sample I/O for clarification.

**Constraint:**

*You are not allowed to use nested loop. Note that calling a function from a loop and the body of the function containing a loop is considered as nested loop too.*

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 5 | 719 |
| 7 | 40319 |

**--------------------------------------------------- E --------------------------------------------------**

Given two integer m and n. You need to find all the prime numbers in between m and n.

**Input**

A line consisting of two integers m and n. 1 <= m, n <= 1000000. you can assume that |n-m| <= 10000.

**Output**

Print all the prime numbers in the given range. Print “empty” if there is no prime number. See the sample I/O for clarification.

**Constraint:**

*You are not allowed to use nested loop. Note that calling a function from a loop and the body of the function containing a loop is considered as nested loop too.*

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 1 10 | 2 3 5 7 |
| 5 60 | 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 |
| 110 150 | 113 127 131 137 139 149 |