1. Armstrong Number Checker

Write a program that takes an integer as input and checks if it is an Armstrong number or not. If the number is an Armstrong number, your program should output "YES"; otherwise, it should output "NO".

An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself. Example: 153 = (1)^3 + (3)^3 + (5)^3 Note: For 4-digit numbers you have to consider sum of the 4th power of its digits, for example: 9474 = (9)^4 + (4)^4 + (7)^4 + (4)^4

Input Format

A single integer 'n' (1 <= n <= 10^9) representing the number to be checked.

Constraints

The input integer 'n' will be positive.

Output Format

Output "YES" if the given number is an Armstrong number. Output "NO" if the given number is not an Armstrong number.

2. Factor Finder

Write a program that takes a positive integer as input and prints all its factors.

Input Format

A single integer 'n' (1 <= n <= 10^9) representing the number for which you need to find the factors.

Constraints

The input integer 'n' will be positive.

Output Format

Print a list of integers, each representing a factor of the given number 'n'. The factors should be printed in ascending order, separated by spaces.

3. Pattern Drawer

Write a C program to draw the following pattern using loops: 1 1 2 1 2 3 1 2 3 ...

Input Format

Integer "N" which will define the number of rows you have to print

Constraints

1

Output Format

Print the pattern as shown above.

4. Uppercase Alphabet ASCII Printer

Write a C program that prints the ASCII values of all uppercase alphabets from 'A' to 'Z' using a goto statement. The program should display each uppercase alphabet along with its ASCII value enclosed in square brackets.

Output Format

Print the uppercase alphabets from 'A' to 'Z' along with their corresponding ASCII values enclosed in square brackets. Separate each alphabet and its ASCII value with a space.

5. Maximum and Minimum Finder

Write a C program that accepts three integers from the keyboard and prints the maximum and minimum values among them.

Input Format

Three integers, 'a', 'b', and 'c', separated by spaces on a single line.

Constraints

(-10^9 <= a, b, c <= 10^9)

Output Format

Two integers separated by a space: the maximum value and the minimum value among 'a', 'b', and 'c'.

6. Olympic Year Checker

Write a program that accepts an integer representing a year and determines whether it is an "Olympic year" according to the given rules. The Olympic Games have been held every four years since 1896 except for the years between 1940 and 1947

Input Format

A single integer 'year' (e.g., 1896, 2024) representing the year to be checked.

Constraints

(-10^9 <= year <= 10^9)

Output Format

Output "YES" if the given year is an Olympic year. Output "NO" if the given year is not an Olympic year.

7. Vowel or Consonant Checker with Switch

Write a C program that takes a character as input, checks if it is an alphabetic character (lowercase or uppercase), and then determines whether it is a vowel or a consonant. Print a suitable message based on the result.

Input Format

A single character 'ch' entered by the user.

Constraints

The input character will be from the ASCII Character Set

Output Format

If the input character is a vowel (a, e, i, o, u) or its uppercase counterpart, print "Vowel."
If the input character is a consonant (not a vowel), print "Consonant."

8. Prime Number Checker

Write a C program that takes a positive integer as input and checks whether it is a prime number or not. You need to print "Prime" if the number is prime and "Not Prime" if it is not.

Input Format

A single positive integer 'n' (2 <= n <= 10^9) representing the number to be checked.

Constraints

The input integer 'n' will be positive.

Output Format

Print "Prime" if the given number is prime. Print "Not Prime" if the given number is not prime.

9. Maximum of Entered numbers

Write a C program that repeatedly takes integer input from the user until the user decides to stop. When the user stops entering numbers, your program should find and display the largest among all the entered numbers.

Input Format

The program should repeatedly prompt the user to enter an integer. To stop entering numbers, the user should input a specific character or number (e.g., 0, -1, 'Q', 'q', etc.).

Constraints

The integers entered can be both positive and negative. The user may choose to stop entering numbers at any point.

Output Format

Print the largest integer among all the numbers entered by the user.

10. Month Days Calendar

Write a C program that takes an integer as input representing a month number (1 to 12) and calculates and displays the number of days in that month. The program should account for leap years.

Input Format

A single integer 'month' representing the month number for which you need to find the number of days.

Constraints

The input integer 'month' will be between 1 and 12.

Output Format

Print an integer representing the number of days in the specified month.

11. LCM Calculator using HCF

Write a C program that takes two positive integers as input and calculates and displays their LCM using their HCF (GCD).

Input Format

Two positive integers 'num1' and 'num2' (1 <= num1, num2 <= 10^9) separated by a space.

Constraints

The input integers 'num1' and 'num2' will be positive.

Output Format

Print an integer representing the LCM of 'num1' and 'num2' calculated using their HCF (GCD).

12. Character Counter

Write a C program that takes an input string from the user and counts the following:

The number of uppercase letters (A to Z). The number of lowercase letters (a to z). The number of other characters (anything that is not an alphabetic character).

Hint: You can iterate over every character of given string and then you can update count of uppercase or lowercase accordingly.

Input Format

A string of characters entered by the user. The string can contain alphabetic and non-alphabetic characters.

Constraints

The input string can have a maximum length of 100 characters.

Output Format

Print three integers separated by spaces representing the count of uppercase letters, lowercase letters, and other characters in the input string.

13.Positive Number Sum Calculator

Write a C program that repeatedly takes integer inputs from the user using a do-while loop. The program should continue taking inputs until a negative number is entered. Once a negative number is entered, the program should calculate and display the sum of all positive numbers entered (excluding the negative number).

Input Format

A sequence of integer inputs, one per line. The input sequence will end when a negative number is entered.

Constraints

The input integers can be both positive and negative. The input sequence will contain at least one positive number before the negative number is entered.

Output Format

Print the sum of all positive numbers entered by the user (excluding the negative number).

14. Square and Cube Printer

Write a C program that takes an integer 'N' as input and prints the square and cube of all numbers from 1 to N using a goto statement.

Input Format

A single positive integer 'N' (1 <= N <= 100).

Constraints

The input will be an integer.

Output Format

For each number from 1 to N, print its square and cube separated by a space on the same line.