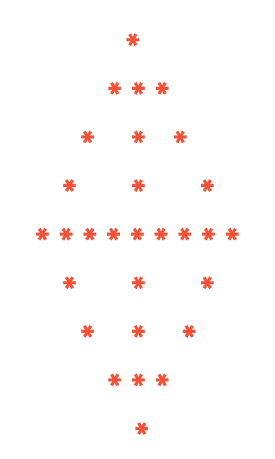
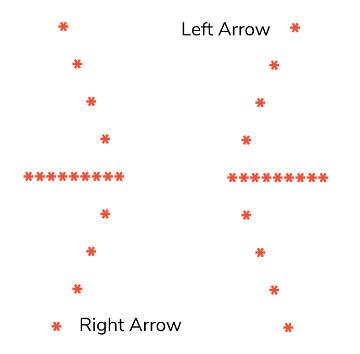
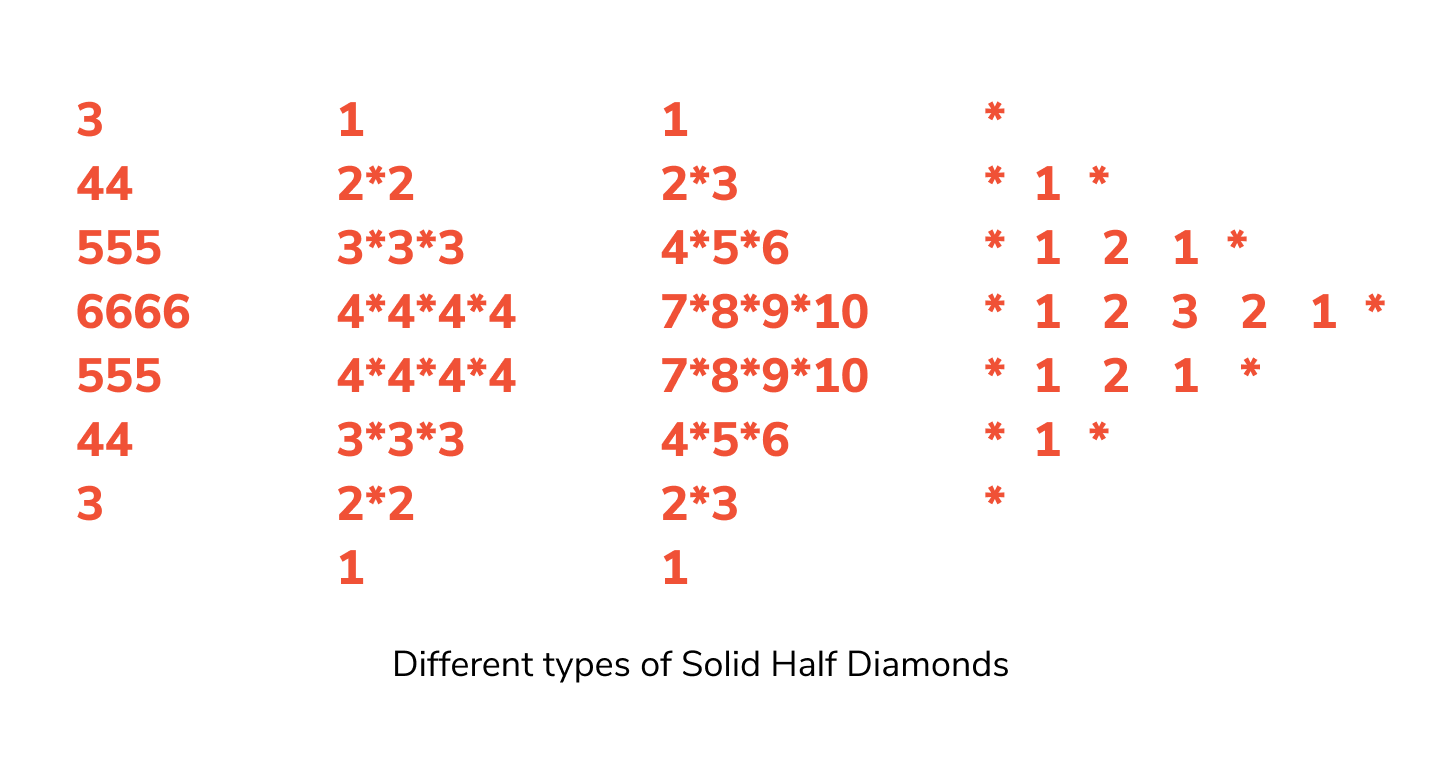
DAY 1:



DAY 2:



DAY 3:



DAY 4:

# Check if two strings match where one string contains wildcard characters

Program to check if two strings match where one string contains wildcard characters is discussed here

**DESCRIPTION:**

Given a text and pattern string. The pattern consists of the following characters

**+:** It can be replaced with 0 or more occurrence of the previous character

**\*:** Matches any sequence of characters (including the empty sequence)

**?:** It can be replaced with a single occurrence of any character.

The task is to determine if the string and pattern match after successfully replacing the special characters in the pattern with the above rules. Print true if the text and pattern match else print false

**Test Cases:**

The first string is the pattern and the second string represents the text

**Sample Input 1:**

String 1: Am?zo

String 2: Amazon

**Sample Output 1:**

FALSE

**Sample Input 2:**

String 1:Am?z\*on

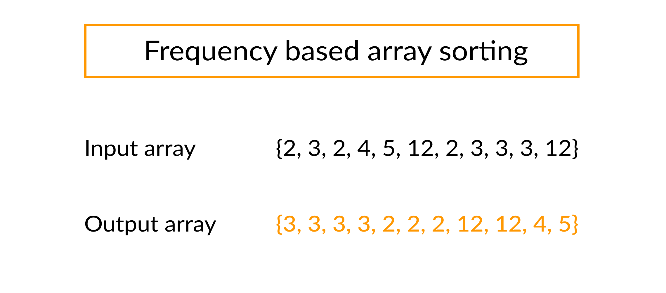
String 2:Amazkfdsaon

**Sample Output 2:**

TRUE

DAY 5:

Program to sort elements by frequency in a given array is discussed here. You need to print the elements of an array in the decreasing frequency and if 2 numbers have same frequency then print the one which came first.

****

DAY 6:

Circular array rotation means rotating the elements in the array where one rotation operation moves the last element of the array to the first position and shifts all remaining elements to the right.

For example, consider the following array = [4, 5, 6], number of rotations = 2, indexes to be checked = [0,1,2]

* Initial array   **[4, 5, 6]**
* After one rotation   **[6, 4, 5]**
* After two rotations   **[5, 6, 4]**

### **OUTPUT**

Element at index 0 : 5

Element at index 1 : 6

Element at index 2 : 4

DAY 7:

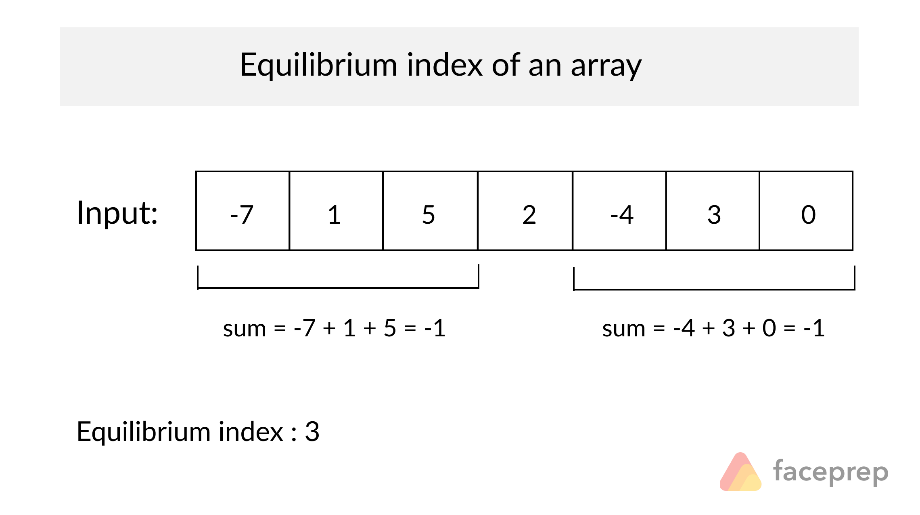
Program to find the equilibrium index of an array is discussed here. Equilibrium index of an array is an index such that

**Sum of elements at lower indexes = Sum of elements at higher indexes.**

For example, consider the array a[] = {-7, 1, 5, 2, -4, 3, 0}. Here, a[0] + a[1] + a[2] = a[4] + a[5] + a[6]. Hence, 3 is the equilibrium index.

## **Method 1:**

* Using two loops.
* Outer loop iterates through all the element and the inner loop check out whether the current index picked by the outer loop is either equilibrium index or not.
* The time complexity of this solution is O(n^2).



DAY 7:

Write a program to find largest possible difference between two prime numbers in the given range

**Input Format**

Accept two integer as a input

**Constraints**

nill

**Output Format**

Print largest possible difference between two prime numbers

**Sample Input 0**

11 25

**Sample Output 0**

12

DAY 8:

Write a program to given a range of numbers ,find the highly composite number in that range.(has the greatest number of factors)

**Input Format**

Accept two integer as a input

**Constraints**

nill

**Output Format**

integer which has more factors between the range

**Sample Input 0**

10 50

**Sample Output 0**

48

DAY 9:

Generate a series where the odd terms are multiples of 2 and even terms are obtained by dividing previous term by 2

**Input Format**

Accept integer as a input

**Constraints**

nill

**Output Format**

Display first n elements of the series

**Sample Input 0**

15

**Sample Output 0**

0 0 2 1 4 2 6 3 8 4 10 5 12 6 14

Day10:

