Merge

using System;

class Program

{

static void Main()

{

int[] A = { 1, 3, 5, 7 };

int[] B = { 2, 4, 6, 8 };

int[] mergedArray = MergeSortedArrays(A, B);

Console.WriteLine("Merged sorted array:");

foreach (int num in mergedArray)

{

Console.Write(num + " ");

}

}

static int[] MergeSortedArrays(int[] A, int[] B)

{

int[] result = new int[A.Length + B.Length];

int i = 0, j = 0, k = 0;

// Use for loops to merge the arrays while there are elements in both

for (; i < A.Length && j < B.Length; k++)

{

if (A[i] <= B[j])

{

result[k] = A[i];

i++;

}

else

{

result[k] = B[j];

j++;

}

}

// Copy any remaining elements of A

for (; i < A.Length; i++, k++)

{

result[k] = A[i];

}

// Copy any remaining elements of B

for (; j < B.Length; j++, k++)

{

result[k] = B[j];

}

return result;

}

}

Non repeating

using System;

using System.Collections.Generic;

class Program

{

static void Main()

{

string input = "Swiss";

char firstNonRepeatingChar = FindFirstNonRepeatingCharacter(input);

if (firstNonRepeatingChar != '\0')

{

Console.WriteLine($"The first non-repeating character in '{input}' is: {firstNonRepeatingChar}");

}

else

{

Console.WriteLine("No non-repeating character found.");

}

}

static char FindFirstNonRepeatingCharacter(string s)

{

// Dictionary to store character counts

Dictionary<char, int> charCount = new Dictionary<char, int>();

// Count occurrences of each character in the string

foreach (char c in s)

{

char lowerChar = char.ToLower(c); // Convert to lowercase for case insensitivity

if (charCount.ContainsKey(lowerChar))

{

charCount[lowerChar]++;

}

else

{

charCount[lowerChar] = 1;

}

}

// Find the first character with count == 1

foreach (char c in s)

{

char lowerChar = char.ToLower(c); // Convert to lowercase for case insensitivity

if (charCount[lowerChar] == 1)

{

return c;

}

}

// If no non-repeating character found, return '\0'

return '\0';

}

}

Missing numbers

using System;

class Program

{

static void Main()

{

int[] A = { 3, 0, 1, 4 };

int missingNumber = FindMissingNumber(A);

Console.WriteLine($"The missing number is: {missingNumber}");

}

static int FindMissingNumber(int[] nums)

{

int n = nums.Length;

int expectedSum = n \* (n + 1) / 2;

int actualSum = 0;

foreach (int num in nums)

{

actualSum += num;

}

return expectedSum - actualSum;

}

}

Armstrong

using System;

class Program

{

static void Main()

{

int number = 153; // Example number

if (IsArmstrongNumber(number))

{

Console.WriteLine($"{number} is an Armstrong number.");

}

else

{

Console.WriteLine($"{number} is not an Armstrong number.");

}

}

static bool IsArmstrongNumber(int num)

{

int originalNum = num;

int sum = 0;

while (num > 0)

{

int digit = num % 10;

sum += digit \* digit \* digit;

num /= 10;

}

return sum == originalNum;

}

}

Common prefix

using System;

class Program

{

static void Main()

{

Console.WriteLine("Enter the number of strings:");

int n = int.Parse(Console.ReadLine());

string[] strs = new string[n];

Console.WriteLine("Enter the strings:");

for (int i = 0; i < n; i++)

{

strs[i] = Console.ReadLine();

}

string longestCommonPrefix = FindLongestCommonPrefix(strs);

if (longestCommonPrefix.Length > 0)

{

Console.WriteLine($"The longest common prefix is: '{longestCommonPrefix}'");

}

else

{

Console.WriteLine("There is no common prefix.");

}

}

static string FindLongestCommonPrefix(string[] strs)

{

if (strs == null || strs.Length == 0)

return "";

string prefix = strs[0];

for (int i = 1; i < strs.Length; i++)

{

while (strs[i].IndexOf(prefix) != 0)

{

prefix = prefix.Substring(0, prefix.Length - 1);

if (prefix.Length == 0)

return "";

}

}

return prefix;

}

}

Fibonacci

using System;

class Program

{

static void Main()

{

Console.WriteLine("Enter the number of terms:");

int n = int.Parse(Console.ReadLine());

PrintFibonacciSequence(n);

}

static void PrintFibonacciSequence(int n)

{

int first = 0, second = 1;

Console.WriteLine("Fibonacci sequence:");

for (int i = 1; i <= n; i++)

{

Console.Write(first + " ");

int next = first + second;

first = second;

second = next;

}

Console.WriteLine();

}

}

Numbers

using System;

class Program

{

static void Main()

{

Console.WriteLine("Enter N:");

int N = int.Parse(Console.ReadLine());

int[] values = new int[N];

Console.WriteLine($"Enter {N} values:");

for (int i = 0; i < N; i++)

{

values[i] = int.Parse(Console.ReadLine());

}

int positiveCount = 0;

int negativeCount = 0;

int total = 0;

foreach (int value in values)

{

if (value > 0)

{

positiveCount++;

}

else if (value < 0)

{

negativeCount++;

}

total += value;

}

double average = (double)total / N;

Console.WriteLine($"The number of positive numbers: {positiveCount}");

Console.WriteLine($"The number of negative numbers: {negativeCount}");

Console.WriteLine($"Total is: {total}");

Console.WriteLine($"Average is: {average}");

}

}