**Question 1**

static void Question1()

{

Console.Write("Enter 16-digit number : ");

string num = Console.ReadLine();

int[] intArr = new int[16];

// insert into array

for (int i = 0; i < 16; i++)

{

intArr[i] = (int)Char.GetNumericValue(num[i]);

}

// reverse

Array.Reverse(intArr);

// multiply even loc by 2 and get sum

for (int i = 1; i < 16; i += 2)

{

int n = intArr[i];

n = n \* 2;

if (n > 9)

n = (n % 10) + 1;

intArr[i] = n;

}

int total = intArr.Sum();

string output;

if (total % 10 == 0)

output = "Card number is valid";

else

output = "Card number is invalid";

Console.WriteLine(output);

}

Graphical user interface, text

Description automatically generated

**Question2**

static void Question2()

{

Console.Write("Enter 11 numbers separated by ',' : ");

string num = Console.ReadLine();

string[] strArr = num.Split(',');

int[] intArr = new int[11];

// insert into int array

for(int i=0; i<11; i++)

{

intArr[i] = Convert.ToInt32(strArr[i]);

}

// find duplicates

for (int i = 0; i < 11; i++)

{

var result = Array.FindAll(intArr, element => element.Equals(i));

if (result.Length == 1)

Console.WriteLine("Non-repeating number : " + Convert.ToString(result[0]));

}

}

**A picture containing graphical user interface

Description automatically generated**

**Question 3**

static void Question3()

{

Console.Write("Enter a number: ");

int input = Convert.ToInt32(Console.ReadLine());

List<int> numlist = new List<int>();

while (input >= 0)

{

Console.Write("Enter another number: ");

input = Convert.ToInt32(Console.ReadLine());

numlist.Add(input);

}

int[] intArr = numlist.ToArray();

Array.Sort(intArr);

// median

int count = intArr.Count();

int midIdx = intArr.Count() / 2;

double median;

if (count % 2 == 0) // odd

median = intArr[midIdx];

else // even

median = (intArr[midIdx] + intArr[midIdx+1]) / 2.0;

Console.WriteLine("Median : " + Convert.ToString(median));

// mode

int curCount = 1;

int maxCount = 1;

int mode = intArr[0];

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

{

if(intArr[i] == intArr[i-1])

{

curCount++;

}

else

{

if (curCount > maxCount)

{

maxCount = curCount;

mode = intArr[i - 1];

}

curCount = 1;

}

}

if (curCount > maxCount)

{

maxCount = curCount;

mode = intArr[count];

}

Console.WriteLine("Mode : " + Convert.ToString(mode));

}

**Graphical user interface, text, chat or text message

Description automatically generated**

**Question 6 (Leetcode)**

public int RemoveDuplicates(int[] nums)

{

int idx = 1;

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

{

if (nums[i] != nums[i - 1])

{

nums[idx] = nums[i];

idx++;

}

}

return idx;

}