31.Armstrong number in given range.

/\*is equal to that number, the number is called Armstrong number. For example 153.\*/

/\*Sum of its divisor is 13 + 53;+ 33; = 1+125+27 = 153\*/

using System;

public class Program31

{

public static void Main()

{

int num,r,sum,temp;

int stno,enno;

Console.Write("\n\n");

Console.Write("Find the Armstrong number for a given range of number:\n");

Console.Write("--------------------------------------------------------");

Console.Write("\n\n");

Console.Write("Input starting number of range: ");

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

Console.Write("Input ending number of range : ");

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

Console.Write("Armstrong numbers in given range are: ");

for(num=stno;num<=enno;num++){

temp=num;

sum = 0;

while(temp!=0){

r=temp % 10;

temp=temp/10;

sum=sum+(r\*r\*r);

}

if(sum==num)

Console.Write("{0} ",num);

}

Console.Write("\n");

}

}

32.Calculate the number of digits in an integer.

using System;

class Program32

{

static void Main(string[] args)

{

Console.Write("\n\n Recursion : Count the number of digits in a number :\n");

Console.Write("---------------------------------------------------------\n");

Console.Write(" Input any number : ");

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

Console.Write("\n The number {0} contains number of digits : {1} ",num,getDigits(num, 0));

Console.ReadLine();

}

public static int getDigits(int n1, int nodigits)

{

if (n1 == 0)

return nodigits;

return getDigits(n1 / 10, ++nodigits);

}

}

33.Find Smallest and largest number from Array.

**using** System;

**class** Program33

{

**static** **void** Main()

{

**int** i;

**int**[] a = **new** **int**[30]; // Array Declaration in C#

Console.Write("Enter the Number of values to find Smallest and Largest Number: ");

**int** n = Convert.ToInt16(Console.ReadLine()); // read the string value and convert it in to integer

Reading the values one by one

**for** (i = 1; i <= n; i++)

{

Console.Write("Enter the No " + i + ":");

a[i] = Convert.ToInt16(Console.ReadLine());

}

**for** (i = 1; i <= n; i++)

{

**for** (**int** j = 1; j <= n - 1; j++)

{

**if** (a[j] > a[j + 1])

{

**int** temp = a[j];

a[j] = a[j + 1];

a[j + 1] = temp;

}

}

}

//Display the Smallest value

Console.WriteLine("The smallest Value is "+a[1]);

//Display the Biggest Value

Console.WriteLine("The Largest Value is " + a[n]);

//Waiting for output

Console.ReadKey();

}

}

34.Find Number of even and odd elements from Array.

**using** System;

**class** Program34

{

**static** **void** CountingEvenOdd(**int**[] arr, **int** arr\_size)

{

**int** even\_count = 0;

**int** odd\_count = 0;

**for** (**int** i = 0; i < arr\_size; i++) {

     // completely divisible by 2

**if** ((arr[i] & 1) == 1)

     odd\_count++;

**else**

     even\_count++;

  }

  Console.WriteLine("Number of even"

                     + " elements = " + even\_count

                     + " Number of odd elements = "

                     + odd\_count);

}

**public** **static** **void** Main()

{

**int**[] arr = { 2, 3, 4, 5, 6 }

**int** n = arr.Length;

  CountingEvenOdd(arr, n);

 }

}

35.Rotation of elements of array-left and right.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

public class Program35

{

public static void Main()

{

int[] nums = {1, 2, 8};

Console.WriteLine("\nArray1: [{0}]", string.Join(", ", nums));

var temp = nums[0];

for (var i = 0; i < nums.Length - 1; i++)

{

nums[i] = nums[i + 1];

}

nums[nums.Length - 1] = temp;

Console.WriteLine("\nAfter rotating array becomes: [{0}]", string.Join(", ", nums));

}

}