ASSIGNMENT 1

DATE: 12-08-2015

1. String Reversal and Palindrome

METHOD 1:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace String\_reverse

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter the string to be reversed");

string original=Console.ReadLine();

string reverse="";

foreach(char c in original)

reverse=c+""+reverse;

Console.WriteLine("REVERSED VALUE IS :" + reverse);

if (original == reverse)

{

Console.WriteLine("It is a Palindrome");

}

else

{

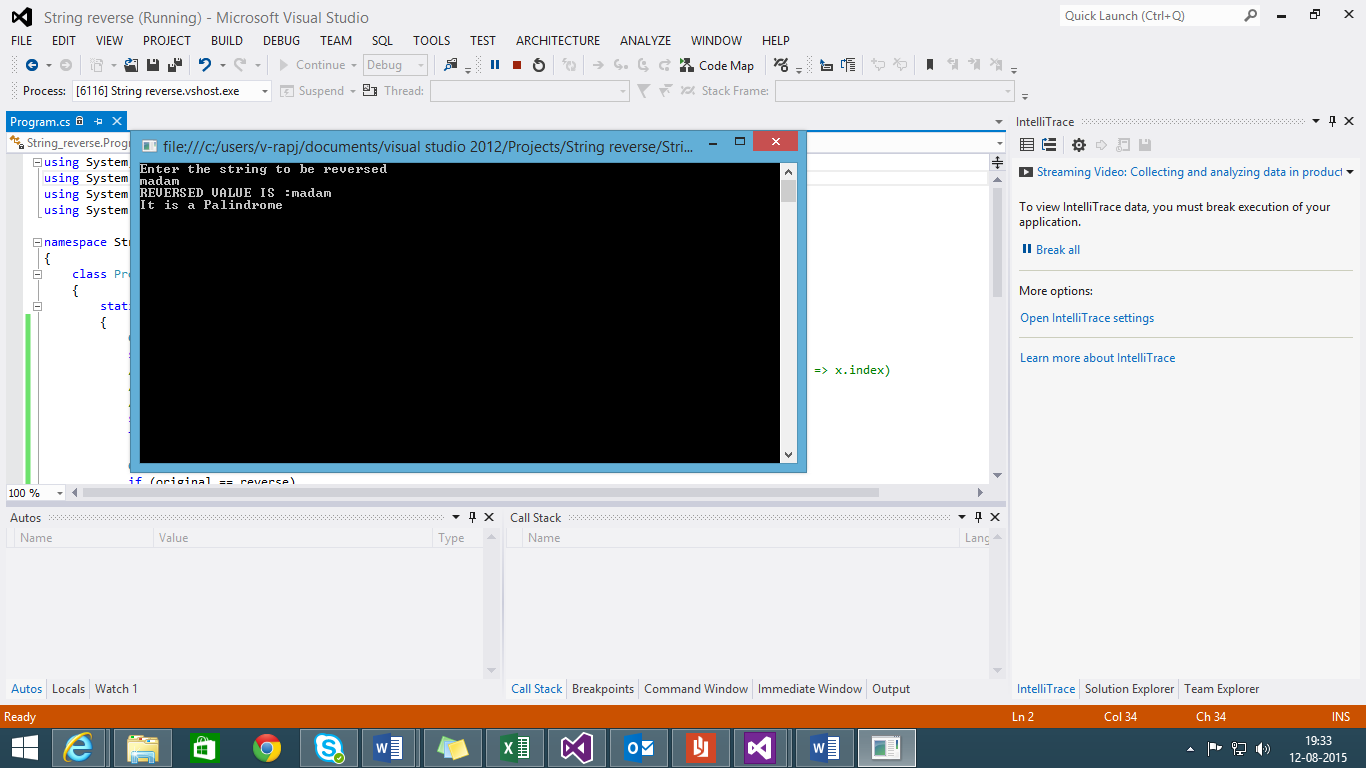
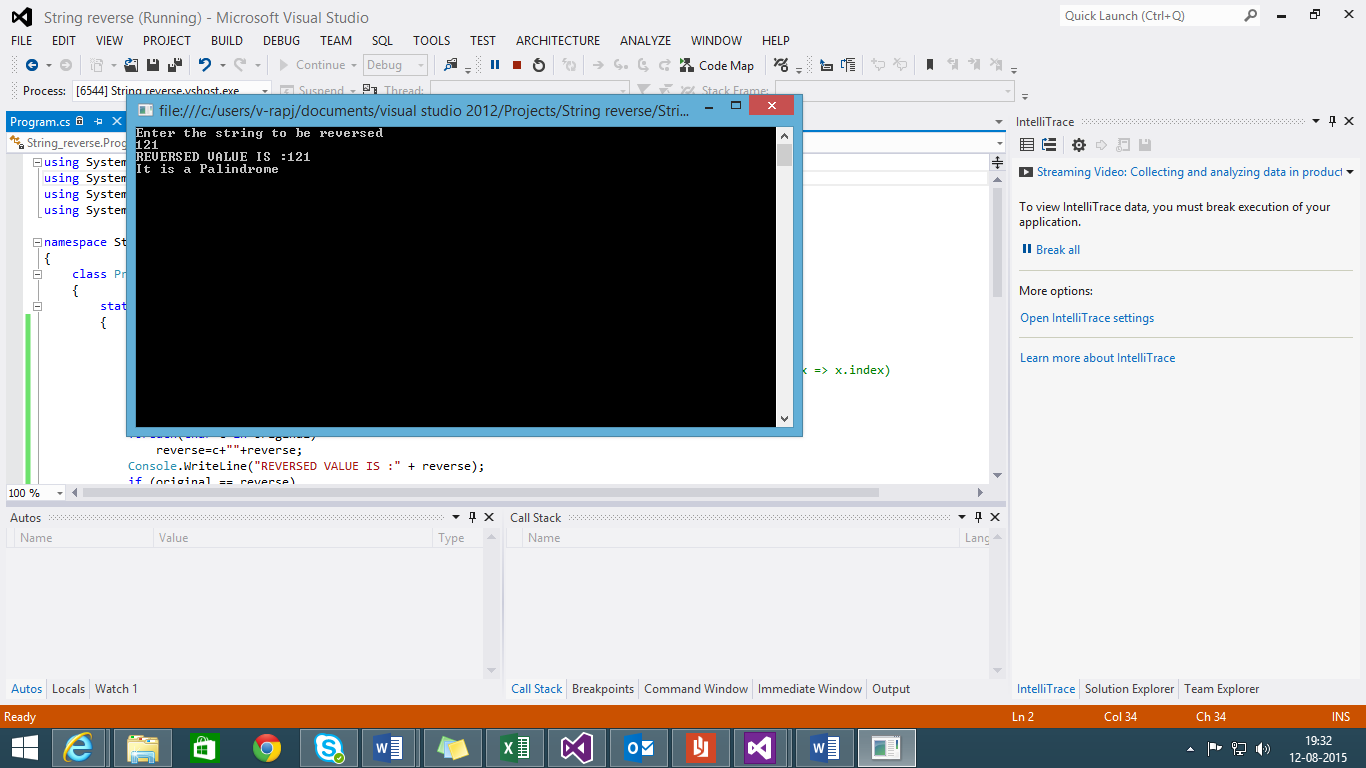
Console.WriteLine("It is not a Palindrome");

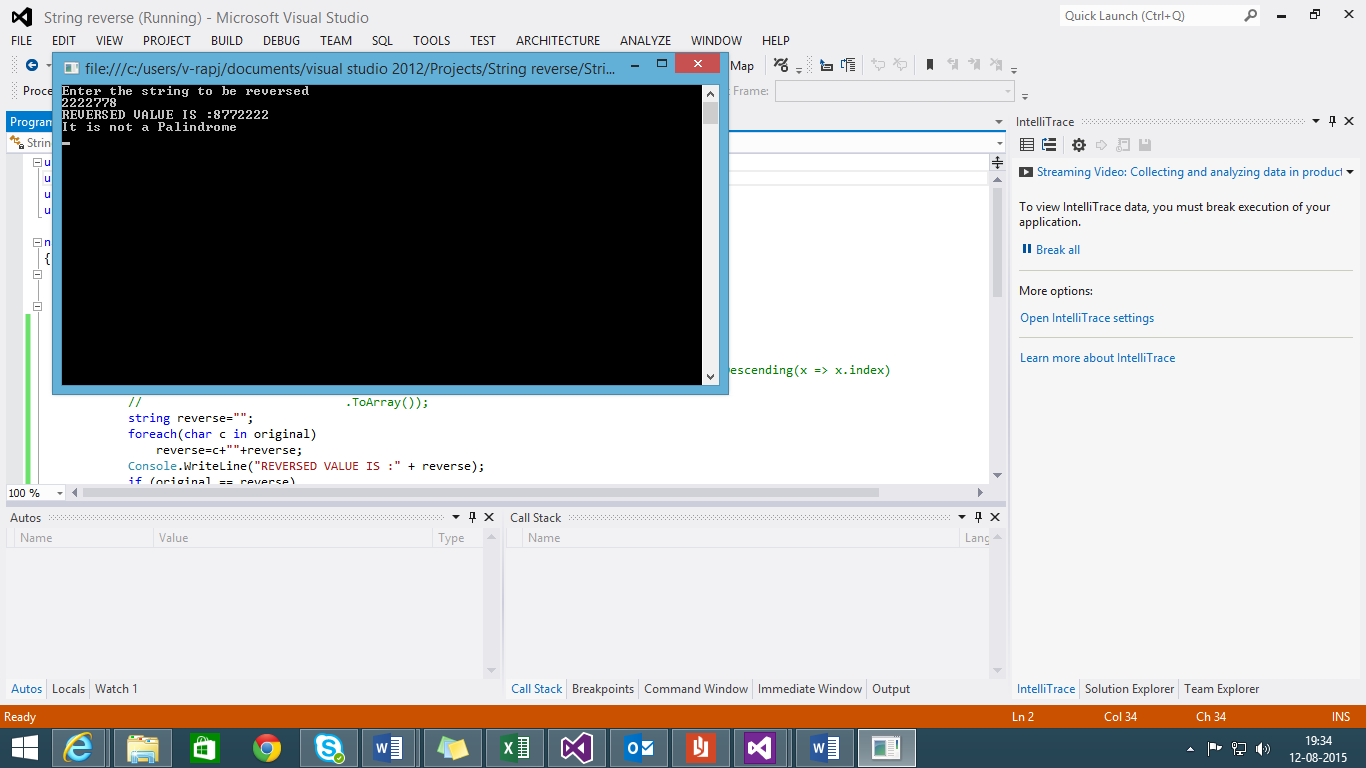
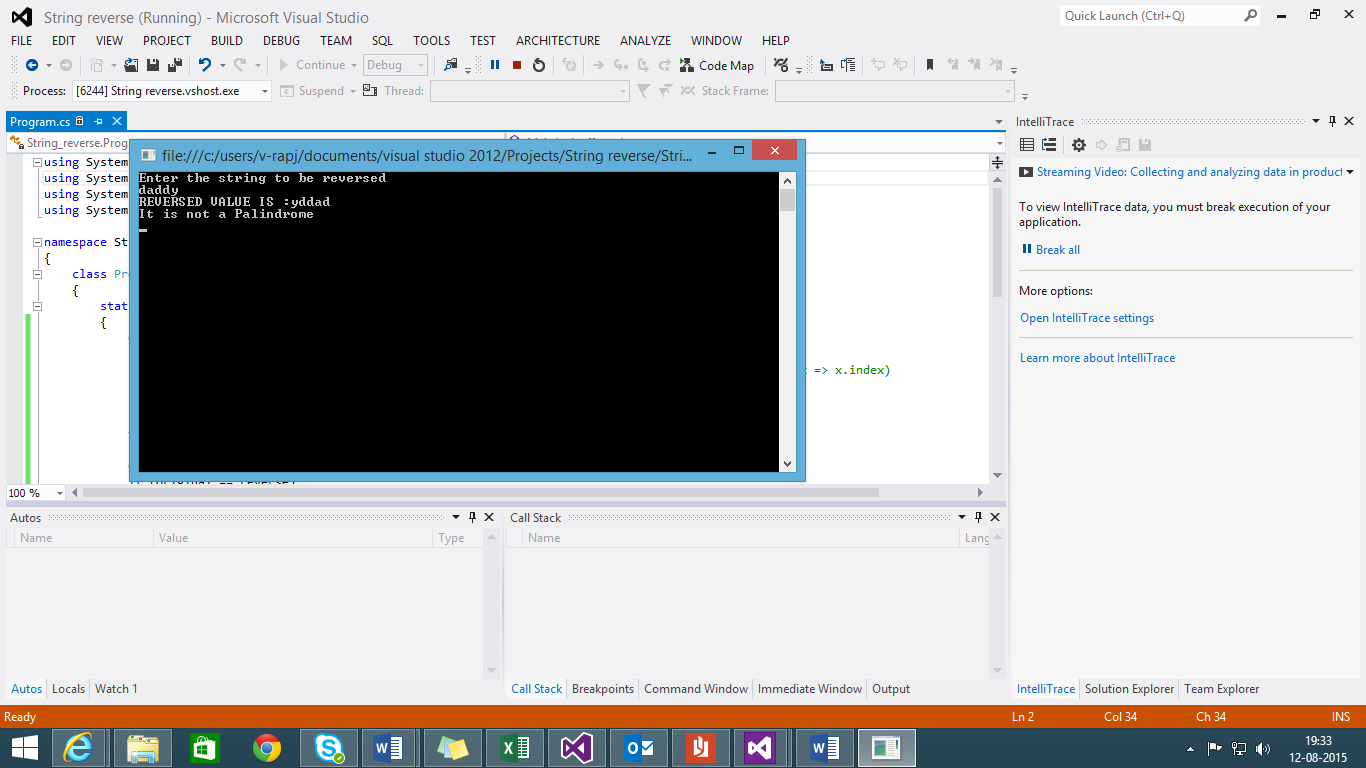
}

Console.ReadKey();

}

}

}

NOT A PALINDROME

METHOD 2:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace String\_reverse

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter the string to be reversed");

string original=Console.ReadLine();

string reverse = "";

for (int i = original.Length - 1; i >= 0; i--)

{

reverse += original[i];

}

Console.WriteLine("REVERSED VALUE IS :" + reverse);

if (original == reverse)

{

Console.WriteLine("It is a Palindrome");

}

else

{

Console.WriteLine("It is not a Palindrome");

}

Console.ReadKey();

}

}

}

METHOD 3:

string reverse = "";

int i=original.Length-1;

while (i >= 0)

{

reverse += original[i];

i--;

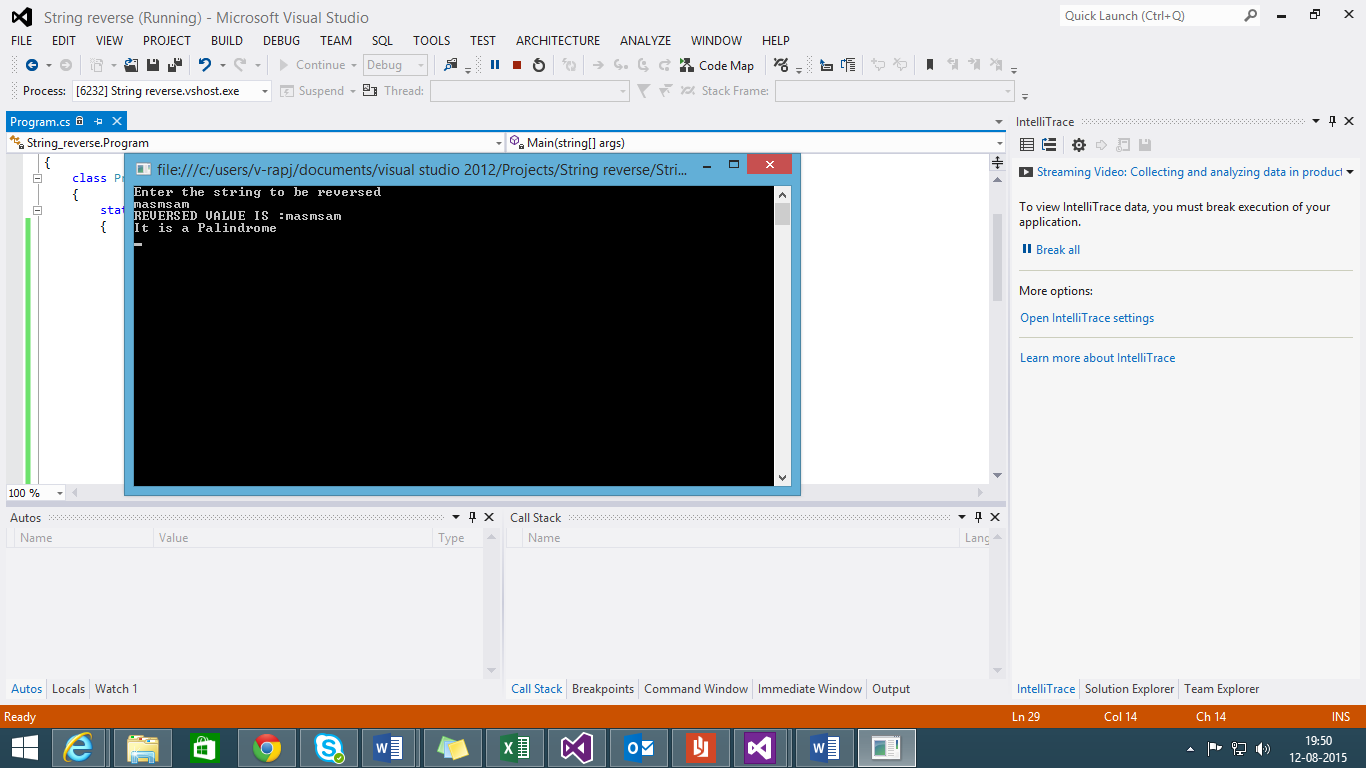
}

Console.WriteLine("REVERSED VALUE IS :" + reverse);

if (original == reverse)

Console.WriteLine("It is a Palindrome");

else

Console.WriteLine("It is not a Palindrome");

METHOD 4:

string reverse = new string(original.Select((c, index) => new{ c, index })

.OrderByDescending(x => x.index)

.Select(x => x.c)

.ToArray());

Using the lambda expression hard to understand but documented for reference

1. Binary to Decimal

METHOD 1:

static void Main(string[] args)

{

Console.WriteLine("Binary to Decimal conversion");

int binary, mod, sum, basic = 1, decim = 0;

for (int i = 0; i < 5; i++)

{

Console.Write("Enter the binary number for conversion:");

binary = Int32.Parse(Console.ReadLine());

sum = binary;

if (sum % 10 == 0 || sum % 10 == 1)

{

while (binary > 0)

{

mod = binary % 10;

decim = decim + (mod \* basic);

binary = binary / 10;

basic = basic \* 2;

}

Console.WriteLine("Binary value is:" + sum);

Console.WriteLine("The decimal value is:" + decim);

}

else

Console.WriteLine("Enter a Valid Binary Number");

basic = 1;

decim = 0;

}

Console.ReadLine();

}

METHOD 2:

Program b = new Program();

int[] a = new int[100];

int n,dec=0;

int j=0;

Console.WriteLine("enter the number of bits for which you want of calculate decimal \n");

n=Int32.Parse(Console.ReadLine());

Console.WriteLine("enter binary number one by one \n");

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

a[k]=Int32.Parse(Console.ReadLine());

for(int l=(n-1);l>=0;l--)

{

dec = (a[l] \* b.power(2,j)) + dec;

j++;

}

Console.WriteLine("binary number of decimal is "+dec);

Method Power

Int power(int c, int d)

{

int pow = 1;

int i = 1;

while (i <= d)

{

pow = pow \* c;

i++;

}

return pow;

}

