**Thực hành Lab 1A**

Thực hiện kiểm thử các chương trình C# để:

* Cho biết lỗi gì xảy ra (có thể giải thích)
* Tô màu dòng lệnh lỗi (tô nền vàng)
* Sửa lỗi xảy ra, nếu có (tô chữ đỏ).

BT 1A/.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text.RegularExpressions;

namespace Test1

{

public class Program

{

public static void Main(string[] args)

{

Console.WriteLine(''Welcome to C# Testing!'');

Console.WriteLine(''Welcome to C# Testing!'');

}

}

}

BT2A/.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text.RegularExpressions;

namespace Test2

{

public class Program

{

public static void Main(string[] args)

{

double number = 1.750.036.490,89;

double number = 1.75003649089;

Console.WriteLine(number);

}

}

}

BT3A/. -> Khi chạy không báo lỗi

using System;

using System.Globalization;

class Class1

{

static void Main(string[] args)

{

Calendar MyCalendar = new GregorianCalendar();

CultureInfo MyCulture = new CultureInfo("es-ES");

DateTime MyDate = new DateTime(2006,8,22,15,30,0,0);

Console.WriteLine(MyCalendar.ToDateTime(MyDate.Year,

MyDate.Month,

MyDate.Day,

MyDate.Hour,

MyDate.Minute, 0, 0));

}

}

BT4A/.

using System;

using System.Globalization;

class Class5

{

static void Main(string[] args)

{

Calendar MyCalendar = new GregorianCalendar();

Calendar MyCalenda = new GregorianCalendar();

Console.WriteLine(MyCalenda.ToFourDigitYear(2020));

Console.WriteLine(MyCalenda.ToFourDigitYear(20));

}

}

BT5A/.

using System;

public class Example3IntDouble {

public static void Main() {

int ivar; // this declares an int variable

double dvar; // this declares a floating-point variable

ivar = 100; // assign ivar the value 100

dvar = 100.0; // assign dvar the value 100.0

Console.WriteLine("Original value of ivar: " + ivar);

Console.WriteLine("Original value of dvar: " + dvar);

Console.WriteLine(); // print a blank line

// now, divide both by 3

ivar = ivar / 3.0;

ivar = Convert.ToInt32(ivar / 3.0);

dvar = dvar / 3.0;

Console.WriteLine("ivar after division: " + ivar);

Console.WriteLine("dvar after division: " + dvar);

}

}

BT6A/.

using System;

public class CompareDemo {

public static void main()

public static void Main() {

string str1 = "one";

string str2 = "one";

string str3 = "ONE";

string str4 = "two";

string str5 = "one, too";

if(String.Compare(str1, str2) == 0)

Console.WriteLine(str1 + " and " + str2 +

" are equal.");

else

Console.WriteLine(str1 + " and " + str2 +

" are not equal.");

if(String.Compare(str1, str3) == 0)

Console.WriteLine(str1 + " and " + str3 +

" are equal.");

else

Console.WriteLine(str1 + " and " + str3 +

" are not equal.");

if(String.Compare(str1, str3, true) == 0)

Console.WriteLine(str1 + " and " + str3 +

" are equal ignoring case.");

else

Console.WriteLine(str1 + " and " + str3 +

" are not equal ignoring case.");

if(String.Compare(str1, str5) == 0)

Console.WriteLine(str1 + " and " + str5 +

" are equal.");

else

Console.WriteLine(str1 + " and " + str5 +

" are not equal.");

if(String.Compare(str1, 0, str5, 0, 3) == 0)

Console.WriteLine("First part of " + str1 + " and " +

str5 + " are equal.");

else

Console.WriteLine("First part of " + str1 + " and " +

str5 + " are not equal.");

int result = String.Compare(str1, str4);

if(result < 0)

Console.WriteLine(str1 + " is less than " + str4);

else if(result > 0)

Console.WriteLine(str1 + " is greater than " + str4);

else

Console.WriteLine(str1 + " equals " + str4);

}

}

BT7A/.

using System;

**class** Rect {

**public** **int** width;

**public** **int** height;

Rect(**int** w, **int** h)

public Rect(**int** w, **int** h) {

width = w;

height = h;

}

**public** **int** area() {

**return** width \* height;

}

}

**public** **class** UseRect {

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

Rect r1 = **new** Rect(4, 5);

Rect r2 = **new** Rect(7, 9);

Console.WriteLine("Area of r1: " + r1.area());

Console.WriteLine("Area of r2: " + r2.area());

}

}

BT8A/.

// declare the Car class

class Car

{

// declare the fields

public string make = "Ford";

public string model = "T";

public string color; // default value of null

public int yearBuilt = 2020;

// define the methods

public void Start()

{

System.Console.WriteLine(model + " started");

}

public void Stop()

{

System.Console.WriteLine(model + " stopped");

}

}

public class Example5\_2

{

public static void Main()

{

// create a Car object

Car myCar = new car();

Car myCar = new Car();

// display the default values for the Car object fields

System.Console.WriteLine("myCar.make = " + myCar.make);

System.Console.WriteLine("myCar.model = " + myCar.model);

if (myCar.color == null)

{

System.Console.WriteLine("myCar.color is null");

}

System.Console.WriteLine("myCar.yearBuilt = " + myCar.yearBuilt);

}

}

BT9A/.

class Car

{

// declare the fields

private string make;

private string model;

private string color;

private int yearBuilt;

// define the constructor

Car(string make, string model, string color, int yearBuilt)

public Car(string make, string model, string color, int yearBuilt)

{

System.Console.WriteLine("In Car() constructor");

this.make = make;

this.model = model;

this.color = color;

this.yearBuilt = yearBuilt;

}

// define a method to display the fields

public void Display()

{

System.Console.WriteLine("Car details:");

System.Console.WriteLine("make = " + make);

System.Console.WriteLine("model = " + model);

System.Console.WriteLine("color = " + color);

System.Console.WriteLine("yearBuilt = " + yearBuilt);

}

}

public class Example9

{

public static void Main()

{

// create a Car object using the constructor

// defined in the class

Car myCar = new Car("Toyota", "MR567", "black", 2020);

// display the values for the Car object fields

myCar.Display();

}

}

BT10A/.

using System;

public class BubbleSort {

public static void Main() {

int[] nums = { 99, -10, 100123, 18, -978,

5623, 463, -9, 287, 49 };

int a, b, t;

int size;

size = 10; // number of elements to sort

// display original array

Console.Write("Original array is:");

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

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

Console.Write(" " + nums[i]);

Console.WriteLine();

// This is the bubble sort.

for(a=1; a < size; a++)

for(b=size-1; b >= a; b--) {

if(nums[b-1] > nums[b]) { // if out of order

// exchange elements

t = nums[b-1];

nums[b-1] = nums[b];

nums[b] = t;

}

}

// display sorted array

Console.Write("Sorted array is:");

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

Console.Write(" " + nums[i]);

Console.WriteLine();

}

}

BT11A/.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

class arrsampl

{

int[,]x;

arrsampl()

{

x = new int[,] { { 11, 2, 61 }, { 42, 50, 3 } };

}

void printarray()

{

Console.WriteLine("Elements in the Given Matrix : ");

for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 3; j++)

{

Console.Write(x[i, j] + "\t");

}

Console.WriteLine("\n");

}

}

int max()

{

int small = x[0, 0];

for (int i = 0; i <= 2; i++)

for (int i = 0; i < 2; i++)

{

for (int j = 0; j <= 3; j++)

for (int j = 0; j < 3; j++)

{

if (small > x[i, j])

{

small = x[i, j];

}

}

}

return small;

}

public static void Main()

{

arrsampl obj = new arrsampl();

obj.printarray();

Console.WriteLine("Smallest Element : {0}", obj.max());

Console.ReadLine();

}

}

BT12A/.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

class arrsampl

{

int[,]x;

arrsampl()

{

x = new int[,] { { 12, 21, 63 }, { 40, 15, 6 } };

}

void printarray()

{

Console.WriteLine("Elements in the Given Matrix : ");

for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 3; j++)

{

Console.Write(x[i, j] + "\t");

}

Console.WriteLine("\n");

}

}

int max()

{

int big = x[0, 0];

for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 3; j++)

{

if (big < x[i, j])

{

big = x[i, j];

}

}

}

return i;

return big;

}

public static void Main()

{

arrsampl obj = new arrsampl();

obj.printarray();

Console.WriteLine("Largest Element : {0}", obj.max());

Console.ReadLine();

}

}