*Câu 18: Xây dựng một lớp đường tròn lưu giữ bán kính và tâm của đường tròn. Tạo các phương thức để tính chu vi, diện tích của đường tròn.*

public class duongtron

{

public double ban\_kinh, x, y;

public double chu\_vi()

{

return Math.PI \* ban\_kinh \* 2;

}

public double dien\_tich()

{

return Math.PI \* ban\_kinh \* ban\_kinh;

}

}

*Câu 19: Thêm thuộc tính BanKinh vào lớp được tạo ra từ bài tập trên.*

public class duongtron

{

public double ban\_kinh, x, y;

public double BanKinh;

public double chu\_vi()

{

return Math.PI \* ban\_kinh \* 2;

}

public double dien\_tich()

{

return Math.PI \* ban\_kinh \* ban\_kinh;

}

}

*Câu 20: Tạo ra một lớp lưu trữ giá trị nguyên tên myNumber. Tạo thuộc tính cho thành viên này. Khi số được lưu trữ (write) thì nhân cho 100. Và khi số được truy cập (read) thì chia cho 100.*

class so\_nguyen

{

private int myNumber\_val;

public int myNumber

{

get

{

return this.myNumber\_val / 100;

}

set

{

this.myNumber\_val = value \* 100;

}

}

}

*Câu 21: Chương trình sau có lỗi. Hãy sửa lỗi của chương trình và biên dịch chương trình. Dòng lệnh nào gây ra lỗi?*

***using System;***

***using System.Console;***

***class VD1***

***{***

***public string first;***

***}***

***class Tester***

***{***

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

***{***

***VD1 vd = new VD1();***

***Write(“Nhap vao mot chuoi: ”);***

***vd.first = ReadLine();***

***Write(“Chuoi nhap vao: {0}”, vd.first);***

***}***

***}***

using System;

class VD1

{

public string first;

}

class Tester

{

public static void Main()

{

VD1 vd = new VD1();

Console.Write("Nhap vao mot chuoi: ");

vd.first = Console.ReadLine();

Console.Write("Chuoi nhap vao: {0}", vd.first);

}

}

*Câu 30 :Hãy xây dựng cây phân cấp các lớp đối tượng sau: Xe\_Toyota, Xe\_Dream, Xe\_Spacy, Xe\_BMW, Xe\_Fiat, Xe\_DuLich, Xe\_May, Xe?*



*Câu 31: Hãy xây dựng các lớp đối tượng trong bài tập trên, thiết lập các quan hệ kế thừa dựa trên cây kế thừa mà bạn xây dựng. Mỗi đối tượng chỉ cần một thuộc tính là myNane để cho biết tên của nó (như Xe\_Toyota thì myName là “Toi la Toyota”...). Các đối tượng có phương thức Who() cho biết giá trị myName của nó. Hãy thực thi sự đa hình trên các lớp đó. Cuối cùng tạo một lớp Tester với hàm Main() để tạo một mảng các đối tượng Xe, đưa từng đối tượng cụ thể vào mảng đối tượng Xe, sau đó cho lặp từng đối tượng trong mảng để nó tự giới thiệu tên (bằng cách gọi hàm Who() của từng đối tượng).*

*Câu 32: Chương trình sau đây có lỗi. Hãy sửa lỗi biên dịch và chạy chương trình. Cho biết lệnh nào gây ra lỗi. Và nguyên nhân gây ra lỗi?*

using System;

**abstract public class Animal**

{

public Animal(string name)

{

this.name = name;

}

// phương thức trừu tượng minh họa việc

// đưa tên của đối tượng

abstract public void Who();

// biến thành viên protected

protected string name;

}

// lớp Dog dẫn xuất từ lớp Animal

**public class Dog : Animal**

{

// hàm khởi dựng lấy hai tham số

public Dog(string name, string color) : base(name)

{

this.color = color;

}

// phủ quyết phương thức trừu tượng Who()

public override void Who( )

{

Console.WriteLine(“Gu gu! Toi la {0} co mau long {1}”, name, color);

}

// biến private của lớp

private string color;

}

**public class Cat : Animal**

{

// hàm khởi dựng lấy hai tham số

public Cat(string name, int weight) : base(name)

{

this.weight = weight;

}

// phủ quyết phương thức trừu tượng Who()

public override void Who( )

{

Console.WriteLine(“Meo meo! Toi la {0} can nang {1}”, name, weight);

}

// biến private của lớp

private int weight;

}

public class Tester

{

static void Main()

{

Animal[] Arr = new Animal[3];

Arr[0] = new Dog(“Lu Lu”, “Vang”);

Arr[1] = new Cat(“Mun”, 5);

Arr[2] = new Animal(“Noname”);

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

{

Arr[i].Who();

}

}

}

Do khai báo ” **abstract**” không thể new được nên chương trình được sửa như sau:

using System;

abstract public class Animal

{

public Animal(string name)

{

this.name = name;

}

// phương thức trừu tượng minh họa việc

// đưa tên của đối tượng

abstract public void Who();

// biến thành viên protected

protected string name;

}

// lớp Dog dẫn xuất từ lớp Animal

public class Dog : Animal

{

// hàm khởi dựng lấy hai tham số

public Dog(string name, string color) : base(name)

{

this.color = color;

}

// phủ quyết phương thức trừu tượng Who()

public override void Who( )

{

Console.WriteLine("Gu gu! Toi la {0} co mau long {1}", name, color);

}

// biến private của lớp

private string color;

}

public class Cat : Animal

{

// hàm khởi dựng lấy hai tham số

public Cat(string name, int weight) : base(name)

{

this.weight = weight;

}

// phủ quyết phương thức trừu tượng Who()

public override void Who( )

{

Console.WriteLine("Meo meo! Toi la {0} can nang {1}", name, weight);

}

// biến private của lớp

private int weight;

}

public class Tester

{

static void Main()

{

Animal[] Arr = new Animal[3];

Arr[0] = new Dog("Lu Lu", "Vang");

Arr[1] = new Cat("Mun", 5);

Arr[2] = new Cat("MiuLe", 7);

//Arr[2] = new Animal("Noname");

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

{

Arr[i].Who();

}

}

}

Câu 39 : Chương trình sau đây có lỗi. Hãy sửa lỗi, biên dịch, và chạy chương trình. Đoạn lệnh nào gây ra lỗi?

**using System;**

**struct TheStruct**

**{**

**public int x;**

**public TheStruct()**

**{**

**x = 10;**

**}**

**}**

**class TestClass**

**{**

**public static void structtaker( TheStruct s)**

**{**

**s.x = 5;**

**}**

**public static void Main()**

**{**

**TheStruct a = new TheStruct();**

**a.x = 1;**

**structtaker( a);**

**Console.WriteLine("a.x = {0}", a.x);**

**}**

**}**

**Chuyến struct thành class:**

using System;

class TheStruct

{

public int x;

public TheStruct()

{

x = 10;

}

}

class TestClass

{

public static void structtaker(TheStruct s)

{

s.x = 5;

}

public static void Main()

{

TheStruct a = new TheStruct();

a.x = 1;

structtaker(a);

Console.WriteLine("a.x = {0}", a.x);

}

}

Câu 40 : Hãy tính kết quả bằng tay mà chương trình sau xuất ra. Sau đó biên dịch và chạy chương trình để đối sánh kết quả.

**using System;**

**class TheClass**

**{**

**public int x;**

**}**

**struct TheStruct**

**{**

**public int x;**

**}**

**class TestClass**

**{**

**public static void structtaker( TheStruct s)**

**{**

**s.x = 5;**

**}**

**public static void classtaker(TheClass c)**

**{**

**c.x = 5;**

**}**

**public static void Main()**

**{**

**TheStruct a = new TheStruct();**

**TheClass b = new TheClass();**

**a.x = 1;**

**b.x = 1;**

**structtaker( a);**

**classtaker(b);**

**Console.WriteLine("a.x = {0}", a.x);**

**Console.WriteLine("b.x = {0}", b.x);**

**}**

**}**

**Kết quả thực hiện chương trình:**

**a.x = 1**

**b.x = 5**

Câu 41 : Viết hàm (phương thức) tính biểu thức với x,y,z∈ℜ.

static public double myfunction(double x;double y;double z )

{

return Math.Sqrt(Math.Sin(x)+Math.Cos(y))/Math.Cos(z);

}

Câu 42 : Hãy định nghĩa phương thức chồng ToString cho lớp CMatrix sao cho với ma trận phức m = (c1,1 c1,2 ... c1,n; ....; cm,1 cm,2 ... cm,n), ci,j=(ri,j,ii,j), Console.Write(m) sẽ in ra m dòng như sau:

r1,1 r1,2 ... r1,n i1,1 i1,2 ... i1,n

...

rm,1 rm,2 ... rm,n im,1 im,2 ... im,n

r1,1 r1,2 ... r1,n i1,1 i1,2 ... i1,n

...

rm,1 rm,2 ... rm,n im,1 im,2 ... im,n

*public override string ToString()*

*{*

*string strMatrix = "(";*

*for (int i = 0; i < nRows; i++)*

*{*

*string str = "";*

*for (int j = 0; j < nCols - 1; j++)*

*{*

*str += matrix[i, j].ToString() + ", ";*

*}*

*str += matrix[i, nCols - 1].ToString();*

*if (i != nRows - 1 && i == 0)*

*strMatrix += str + "\n";*

*else if (i != nRows - 1 && i != 0)*

*strMatrix += " " + str + "\n";*

*else*

*strMatrix += " " + str + ")";*

*}*

*return strMatrix;*

*}*

*public override bool Equals(object obj)*

*{*

*return (obj is CMatrix) && this.Equals((CMatrix)obj);*

*}*

*public bool Equals(CMatrix cm)*

*{*

*return matrix == cm.matrix;*

*}*

*public override int GetHashCode()*

*{*

*return matrix.GetHashCode();*

*}*

Câu 51 : Viết một chương trình tạo một mảng một chiều nguyên chứa giá trị ngẫu nhiên.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace bai51

{

class Program

{

static void Main(string[] args)

{

int Min = 0;

int Max = 20;

// this declares an integer array with 5 elements

// and initializes all of them to their default value

// which is zero

int[] test2 = new int[5];

Random randNum = new Random();

for (int i = 0; i < test2.Length; i++)

{

test2[i] = randNum.Next(Min, Max);

}

}

}

}

Câu 52 : Sắp xếp các thành phần trong mảng theo thứ tự tăng dần và hiển thị kết quả. Làm tương tự với trường hợp sắp xếp mảng theo thứ tự giảm dần.

Sắp xếp tăng dần:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace bai52

{

class Program

{

static void Main(string[] args)

{

int Min = 0;

int Max = 20;

// this declares an integer array with 5 elements

// and initializes all of them to their default value

// which is zero

int[] test2 = new int[5];

Random randNum = new Random();

for (int i = 0; i < test2.Length; i++)

{

test2[i] = randNum.Next(Min, Max);

}

Array.Sort(test2);

for (int i = 0; i < test2.Length; i++)

{

Console.Write(test2[i]);

Console.Write(' ');

}

Console.ReadLine();

}

}

}

Sắp xếp giảm dần:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace bai52

{

class Program

{

static void Main(string[] args)

{

int Min = 0;

int Max = 20;

// this declares an integer array with 5 elements

// and initializes all of them to their default value

// which is zero

int[] test2 = new int[5];

Random randNum = new Random();

for (int i = 0; i < test2.Length; i++)

{

test2[i] = randNum.Next(Min, Max);

}

//Array.Sort(test2); //them dong nay de sap xep tang dan

Array.Sort(test2, //them dong nay de sap xep giam dan

new Comparison<int>(

(i1, i2) => i2.CompareTo(i1)

));

for (int i = 0; i < test2.Length; i++)

{

Console.Write(test2[i]);

Console.Write(' ');

}

Console.ReadLine();

}

}

}

Câu 53 : Viết một chương trình tạo một mảng một chiều nguyên chứa giá trị ngẫu nhiên. Sắp xếp chúng theo thứ tự số âm thì tăng còn số dương thì giảm dần. Hiển thị kết quả ra màn hình.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace bai52

{

class Program

{

static void Main(string[] args)

{

int Min = -20;

int Max = 20;

// this declares an integer array with 5 elements

// and initializes all of them to their default value

// which is zero

int[] randArray = new int[10];

int[] negArray = new int[10];

int[] posArray = new int[10];

int neg\_size, pos\_size;

Random randNum = new Random();

neg\_size = 0; pos\_size = 0;

for (int i = 0; i < randArray.Length; i++)

{

randArray[i] = randNum.Next(Min, Max);

// Tach mang am duong

if (randArray[i] < 0)

{

negArray[neg\_size] = randArray[i];

neg\_size += 1;

}

else

{

posArray[pos\_size] = randArray[i];

pos\_size += 1;

}

}

//Sap xep tang dan mang am

Array.Sort(negArray);

//Sap xep giam dan mang duong

Array.Sort(posArray,

new Comparison<int>(

(i1, i2) => i2.CompareTo(i1)

));

//Ghep hai mang

//Ghep mang am

for (int i = 0; i < negArray.Length; i++)

{

randArray[i] = negArray[i];

}

//Ghep mang duong

for (int i = neg\_size; i < randArray.Length; i++)

{

randArray[i] = posArray[i - neg\_size];

}

//In mang ra man hinh

for (int i = 0; i < randArray.Length; i++)

{

Console.Write(randArray[i]);

Console.Write(' ');

}

Console.ReadLine();

}

}

}

Câu 54 : Viết một chương trình tìm số lớn nhất và nhỏ nhất trong mảng hai chiều có kích thước cố định. Các thành phần của mảng được phát sinh ngẫu nhiên.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace bai54

{

class Program

{

static void Main(string[] args)

{

int max = 1000, min = -1000;

int max\_value, min\_value;

int m, n; // kich thuoc mang hai chieu

int i, j; //bien dem

int[,] ranArray = new int[10,10]; // khai bao mang hai chieu

Random ran\_num = new Random();

m = 10;

n = 10;

// Tao mang hai chieu ngau nhien

for (i = 0;i<m;i++)

{

for (j = 0;j<n;j++)

{

ranArray[i, j] = ran\_num.Next(min,max);

}

}

// Tim so lon nhat va nho nhat

max\_value = ranArray[0,0];

min\_value = ranArray[0,0];

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

if (min\_value > ranArray[i, j])

{

min\_value = ranArray[i, j];

}

if (max\_value < ranArray[i, j])

{

max\_value = ranArray[i, j];

}

}

}

Console.WriteLine("Gia tri lon nhat cua mang la: {0}", max\_value);

Console.WriteLine("Gia tri nho nhat cua mang la: {0}", min\_value);

}

}

}

Câu 55 :Viết chương trình cộng hai ma trận nxm, tức là mảng hai chiều có kích thước n dòng, m cột. Các giá trị của hai mảng phát sinh ngẫu nhiên, cho biết kết quả cộng hai ma trận?

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace bai55

{

class Program

{

public static void print\_array(int[,] arr)

{

int i,j;

int height = arr.GetLength(0);

int width = arr.GetLength(1);

for (i = 0; i < height; i++)

{

for (j = 0; j < width; j++)

{

Console.Write("{0,5}",arr[i,j]);

}

Console.WriteLine();

}

}

static void Main(string[] args)

{

int max = 1000, min = -1000;

int m, n; // kich thuoc mang hai chieu

int i, j; //bien dem

int[,] a = new int[20, 10]; // khai bao mang hai chieu

int[,] b = new int[20, 10]; // khai bao mang hai chieu

int[,] c = new int[20, 10]; // khai bao mang hai chieu

Random ran\_num = new Random();

m = a.GetLength(0);

n = a.GetLength(1);

// Tao mang hai chieu ngau nhien

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

a[i, j] = ran\_num.Next(min, max);

b[i, j] = ran\_num.Next(min, max);

//Tong la mang c

c[i, j] = a[i, j] + b[i, j];

}

}

Console.WriteLine("Mang a: ");

print\_array(a);

Console.WriteLine("Mang b: ");

print\_array(b);

Console.WriteLine("Tong mang a va b: ");

print\_array(c);

Console.ReadLine();

}

}

}