Задание 1

Написати статичний метод ChangeMatrix, що виконує вказані дії над елементами матриці дійсних чисел (матрицю заповнити рандомно або з файла!! в Main()).

Формальні параметри : матриця дійсних чисел, задана дія – об'єкт Action/Func.

Використовуючи написаний метод:

* вивести матрицю на екран;
* вивести на екран позитивні (>=0) елементи матриці;
* збільшити в три рази всі позитивні елементи матриці.

Після виклику останньої дії – вивести змінену матрицю на екран.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace \_8.\_3

{

class Program

{

static void Main(string[] args)

{

int n;

Console.WriteLine("Введите размер матрицы");

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

int[,] A = new int[n, n];

Random rnd = new Random();

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

{

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

{

A[i, j] = rnd.Next(-20, 20);

}

}

Console.WriteLine("Элементы матрицы A:");

MethodAct(A, Show);

Console.WriteLine("Позитивные элементы матрицы A:");

MethodAct(A, ShowPositive);

Console.WriteLine("Видоизмененная матрица A (позитивные элементы умноженны на 3):");

MethodFunc(A, Mult3);

MethodAct(A, Show);

Console.ReadKey();

}

static void MethodAct(int[,] arr, Action<int> act)

{

if (arr == null)

throw new NullReferenceException();

for(int i=0; i<arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

{

act(arr[i, j]);

}

Console.WriteLine();

}

}

static void MethodFunc(int[,] arr, Func<int, int> act)

{

if (arr == null)

throw new NullReferenceException();

for (int i = 0; i < arr.GetLength(0); i++)

{

for (int j = 0; j < arr.GetLength(1); j++)

{

arr[i, j] = act(arr[i, j]);

}

}

}

public static void Show(int num)

{

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

}

public static void ShowPositive(int num)

{

if (num >= 0) Console.Write("{0}\t", num);

else Console.Write("-\t");

}

public static int Mult3(int num)

{

if (num > 0) return num \* 3;

else return num;

}

}

}

Задание 2

Написати статичний метод ReadFiles, що виконує вказані дії над вмістом файлу.

Формальні параметри : файл, задана дія – об'єкт Action/Func.

Використовуючи написаний метод:

* вивести вміст файлу на екран;
* вивести на екран всі числа! (числа складаються с цифр);
* замінити у файлі всі коми/крапки/\*/() на пробіли.

Після виклику останньої дії – вивести вміст файлу на екран.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace Task2

{

class Program

{

static void Main(string[] args)

{

//действия над содержимым файла

Console.WriteLine("Содержимое файла <Task2.txt>:");

ReadFile("C:\\Users\\Acer\\Desktop\\Учеба\\2 курс\\C#\\Лабы\\first.txt", Show);

Console.WriteLine("\nЧисла файла <Task2.txt>:");

ReadFile("C:\\Users\\Acer\\Desktop\\Учеба\\2 курс\\C#\\Лабы\\first.txt", Numbers);

Console.WriteLine("\n\nВидоизмененный вид файла <Task2.txt>:");

ReadFile("C:\\Users\\Acer\\Desktop\\Учеба\\2 курс\\C#\\Лабы\\first.txt", Change);

ReadFile("C:\\Users\\Acer\\Desktop\\Учеба\\2 курс\\C#\\Лабы\\first.txt", Show);

Console.WriteLine();

Console.ReadKey();

}

public static void ReadFile(String path, Action <String> act)

{

act(path);

}

public static void Show(String path)

{

StringBuilder info = new StringBuilder();

string line;

using (StreamReader MyFile = new StreamReader(path))

{

while((line = MyFile.ReadLine())!=null)

{

info.Append(line);

}

Console.WriteLine(info);

}

}

public static void Numbers(String path)

{

string line;

using (StreamReader MyFile = new StreamReader(path))

{

while ((line = MyFile.ReadLine()) != null)

{

foreach (char x in line)

{

if (Char.IsDigit(x)) Console.Write("{0} ", x);

}

}

}

}

//замена ",/./\*/()" на пробелы

public static void Change(String path)

{

StringBuilder file = new StringBuilder();

string line;

using (StreamReader MyFile = new StreamReader(path))

{

while ((line = MyFile.ReadLine()) != null)

file.Append(line);

}

file.Replace('.', ' ');

file.Replace(',', ' ');

file.Replace('(', ' ');

file.Replace(')', ' ');

file.Replace('\*', ' ');

using (StreamWriter MyFile = new StreamWriter(path))

{

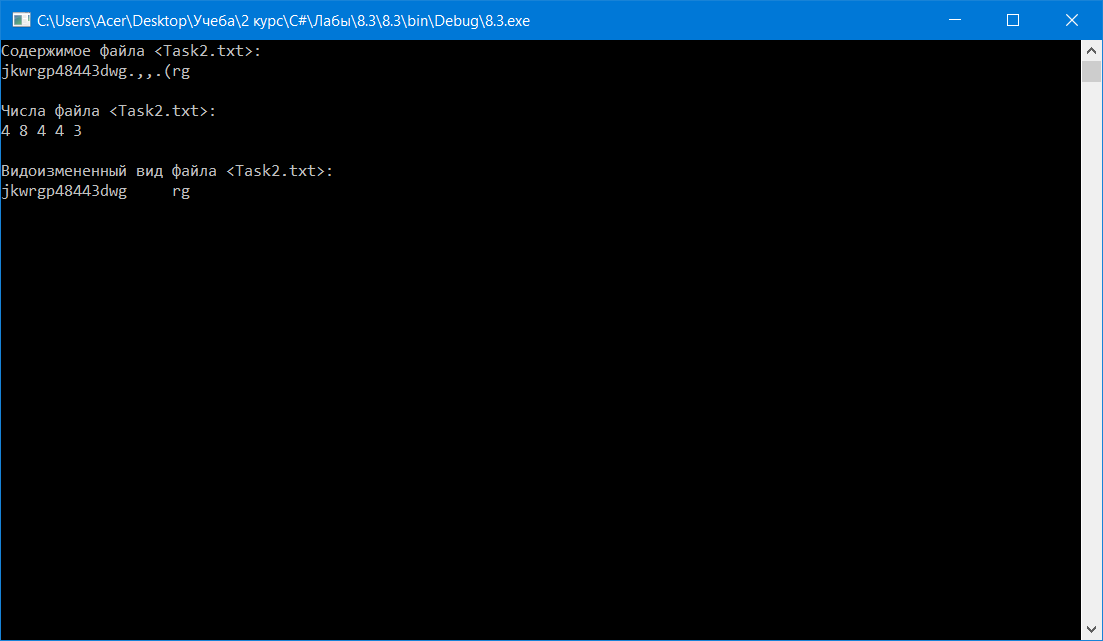
MyFile.WriteLine(file);

}

}

}

}



Завдання 3

Оголосити тип делегату, що посилається на метод. Вимоги до сигнатури методу наступні:

* метод отримує вхідним параметром змінну типу double;
* метод повертає значення типу double, яке є результатом обчислення згідно з умовою задачі.

Реалізувати виклик трьох методів з допомогою одного делегату, які отримують радіус кола R в якості вхідного параметру і обчислюють:

* довжину кола за формулою D = 2πR;
* площу круга за формулою S = πR2;
* об‘єм кулі. Формула V = 4/3πR3.

Методи повинні бути представлені в окремому класі як нестатичні.

Класс Programm

using System;

namespace Task3

{

class Program

{

delegate double Operation(double R);

static void Main(string[] args)

{

double R, result;

Circle function = new Circle();

Operation op;

Console.WriteLine("Введите значение радиуса R:");

R = double.Parse(Console.ReadLine());

if (R <= 0)

throw new FormatException("Радиус должен быть >= 0!");

op = function.Length;

result = op(R);

Console.WriteLine("Длинна круга = {0}, а его радиус = {1}", result, R);

op = function.Square;

result = op(R);

Console.WriteLine("Площадь круга = {0}, а его радиус = {1}", result, R);

op = function.V;

result = op(R);

Console.WriteLine("Обьем сферы = {0}, а его радиус = {1}", result, R);

Console.ReadKey();

}

}

}

Класс Circle

using System;

namespace Task3

{

class Circle

{

public double Length(double R)

{

return 2 \* Math.PI \* R;

}

public double Square(double R)

{

return Math.PI \* R \* R;

}

public double V(double R)

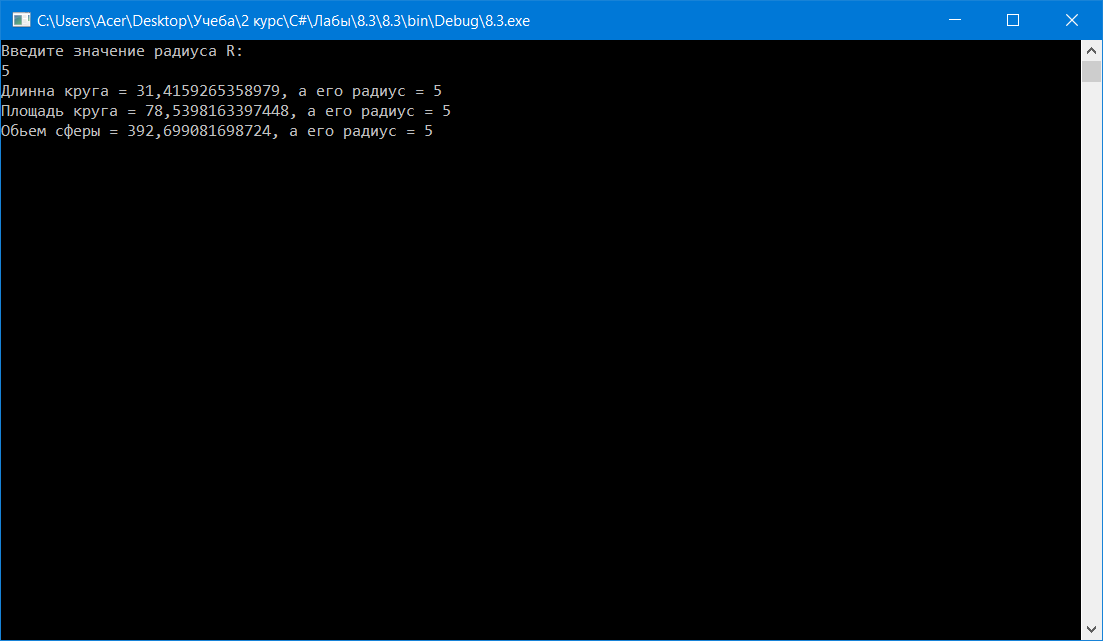
{

return 4 / 3 \* Math.PI \* Math.Pow(R, 3);

}

}

}



Завдання 4

Розібрати приклад Постановка задачі – Створити колекцію, ключем в якій буде значення рядка, а значенням –метод/делегат

using System;

using System.Collections.Generic;

namespace Task4

{

class Program

{

delegate int Operation(int a, int b);

//основаня функция программы

static void Main(string[] args)

{

Dictionary<String, Operation> op = new Dictionary<String, Operation>();

op["add"] = (a, b) => a + b;

op["mult"] = (a, b) => a \* b;

Console.WriteLine(op["add"](5, 4));

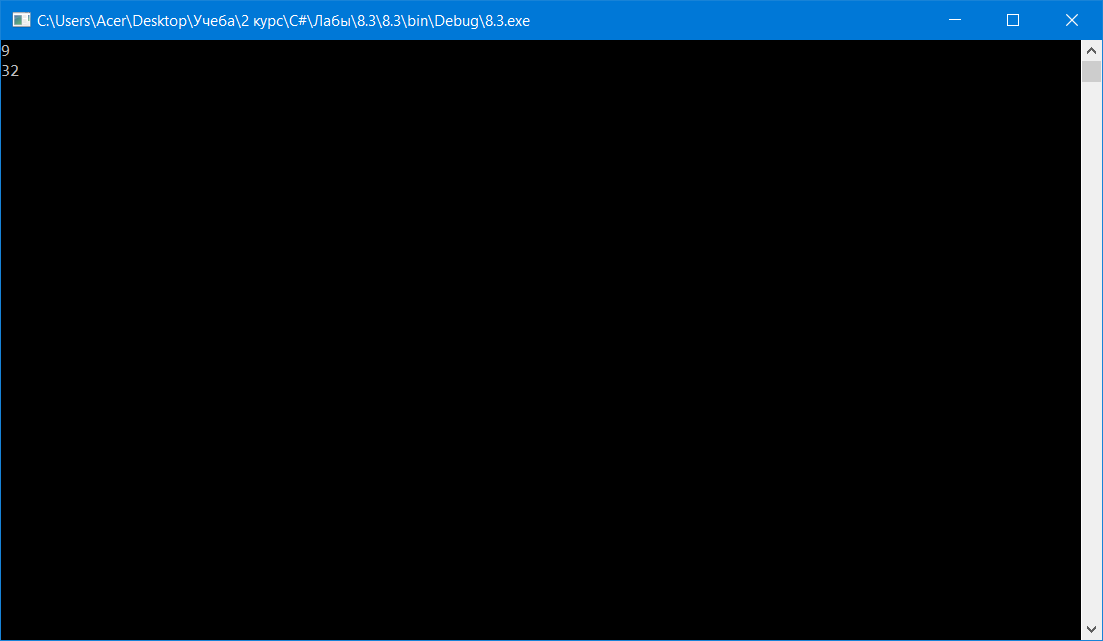
Console.WriteLine(op["mult"](8, 4));

Console.ReadLine();

}

}

}



Задание 5

Оголосіть два делегати: UseOperation, GetGreeting.

Перший делегат посилається на функцію, яка в якості параметрів приймає два значення типу int і повертає деяке ціле число.

Другий делегат посилається на метод без параметрів, який нічого не повертає.

* Створіть методи без параметрів GoodMorning, GoodDay, GoodEvening і GoodNight, при виклику яких виводиться текст «Доброго ранку!», «Добрий день!», «Добрий вечір!» і «Доброї ночі!». Якщо відомо, що властивість DateTime.Now.Hour повертає кількість годин поточного часу, то напишіть програму, в якій, за допомогою делегата GetGreeting, запускається один з написаних вами методів, в залежності від поточного значення часу.
* Створіть методи для обчислення суми, різниці, добутку, частки двох цілих чисел.

Використовуючи делегат UseOperation для виклику описаних в попередньому пункті методів, напишіть програму, яка запитує у користувача два цілих числа і операцію, яку необхідно провести. Виводить відповідний результат.

using System;

using System.Collections.Generic;

namespace Task5

{

class Program

{

public delegate void GetGreeting();

public delegate double UseOperation(double x, double y);

static void Main(string[] args)

{

//Определение времени

int time = DateTime.Now.Hour;

GetGreeting Time;

if (time >= 5 && time <= 10)

Time = GoodMorning;

else if (time > 10 && time <= 17)

Time = GoodDay;

else if (time > 17 && time <= 22)

Time = GoodEvening;

else Time = GoodNight;

Time();

double X, Y;

string operation;

Dictionary<string, UseOperation> op = new Dictionary<string, UseOperation>();

op["+"] = (x, y) => x + y;

op["-"] = (x, y) => x - y;

op["\*"] = (x, y) => x \* y;

op["/"] = (x, y) => x / y;

do

{

Console.WriteLine("Введите нужную Вам операцию (+, -, \*, /):");

operation = Console.ReadLine();

if (op.ContainsKey(operation))

{

break;

}

Console.WriteLine("Операция недоступна!");

} while (true);

Console.WriteLine("Введите параметры X и Y:");

X = double.Parse(Console.ReadLine());

Y = double.Parse(Console.ReadLine());

Console.WriteLine("Результат:");

Console.WriteLine("X " + operation + " Y = " + op[operation](X, Y) + ";");

Console.ReadKey();

}

public static void GoodMorning()

{ Console.WriteLine("Доброе утро!"); }

public static void GoodDay()

{ Console.WriteLine("Добрый день!"); }

public static void GoodEvening()

{ Console.WriteLine("Добрый вечер!"); }

public static void GoodNight()

{ Console.WriteLine("Ложись спать!"); }

}

}

