**ЗАДАНИЕ 20**

Задание 1. Создайте и реализуйте метод решения задачи и выполните

его в объектах класса Task используя три варианта создания объектов класса

Task: Дано четырехзначное число. Найти число, образуемое при

перестановке первой и последней цифр.

Листинг программы:

using System;

class Task

{

private int number;

public Task(int number)

{

this.number = number;

}

public int RearrangeFirstAndLastDigits()

{

if (number < 1000 || number >= 10000)

{

throw new ArgumentOutOfRangeException("number", "The number must be a four-digit number.");

}

int firstDigit = number / 1000;

int lastDigit = number % 10;

return (lastDigit \* 1000) + (number % 1000);

}

}

class Program

{

static void Main(string[] args)

{

Task task1 = new Task(1234);

Task task2 = new Task(task1.RearrangeFirstAndLastDigits());

int inputNumber = 4321;

Task task3 = new Task(inputNumber);

int rearrangedNumber1 = task1.RearrangeFirstAndLastDigits();

int rearrangedNumber2 = task2.RearrangeFirstAndLastDigits();

int rearrangedNumber3 = task3.RearrangeFirstAndLastDigits();

// Print the results

Console.WriteLine($"Rearranged number for Task1: {rearrangedNumber1}");

Console.WriteLine($"Rearranged number for Task2: {rearrangedNumber2}");

Console.WriteLine($"Rearranged number for Task3: {rearrangedNumber3}");

}

}

Таблица 20.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  |  |

Анализ результатов:

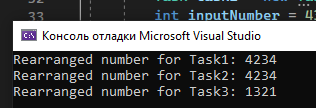


Рисунок 20.1 – Результат работы программы

Задание 2. Создайте массив из 2 задач (объектов класс Task) в каждом

объекте выполните вычисление значения функций и выполните условия:

1. Дождитесь выполнения всех задач;

2. Дождитесь выполнения хот бы одной задачи

Замедлить выполнение задачи можно с помощью Thread.Sleep(n) в

методе, выполняемом задачей; где n – время в миллисекундах.

Листинг программы:

namespace zad2

{

class Program

{

static void Main()

{

Task[] tasks = new Task[2];

tasks[0] = Task.Run(() =>

{

Thread.Sleep(2000);

double alpha = 0.5;

double z1 = (Math.Cos(alpha) + Math.Cos(2\* alpha) + Math.Cos(6 \* alpha) + +Math.Cos(7 \* alpha));

Console.WriteLine("Задача 1 завершена, z1 = " + z1);

});

tasks[1] = Task.Run(() =>

{

Thread.Sleep(1000);

double alpha = 0.5;

double z2 = (4 \* Math.Cos(alpha/2) \* Math.Cos((5/ 2) \* alpha) \* Math.Cos(4 \* alpha ));

Console.WriteLine("Задача 2 завершена, z2 = " + z2);

});

Task.WhenAll(tasks).Wait();

Console.WriteLine("Все задачи завершены");

}

}

}

Таблица 20.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | z2 = -0,8714208234439202  z1 = -0,5085643161327292 |

Анализ результатов:

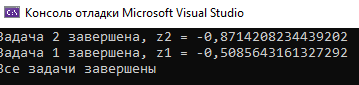


Рисунок 20.2 – Результат работы программы