Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.03 «Программирование в компьютерных системах»

**КВАЛИФИКАЦИОННЫЙ ЭКЗАМЕН**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

*(наименование профессионального модуля)*

Выполнил:

обучающийся группы № 3ПКС-420

*\_\_\_А. М. Лебедева\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(И.О. Фамилия)*

Проверили:

Г.И. Киреева

*(И.О. Фамилия)*

И.В.Сибирев

*(И.О. Фамилия)*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Москва**

**2022**

**Лебедева Александра Михайловна, пкс-420, Экзаменационный билет №2**

**Цель:** освоить работу с классами и коллекциями на языке С#, освоить вставку на языке Assembler.

**Код программы на С#:**

using System;

using System.Collections.Generic;

using System.Linq;

using Binarysharp.Assemblers.Fasm;

using Process.NET;

using Process.NET.Native.Types;

using System.Runtime.InteropServices;

using System.Security;

using System.IO;

namespace Билет\_\_2\_Задание\_1\_пкс420\_Лебедева

{

public class Worker //класс Сотрудник

{

public string Surname; //фамилия сотрудника

public Dictionary<int, int> Month\_salary = new Dictionary<int, int>(); //коллекция зарплат по номеру месяца

public int summMounth\_6 = 0; //зарплата за 6 месяцев

public int avgSumm = 0; //средняя зарплата сотрудника

public Worker (string surname)

{

Surname = surname;

}

public Worker(string surname, string str\_salary)

{

Surname = surname;

string[] str = str\_salary.Split(',');

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

{

Month\_salary.Add(i + 1, int.Parse(str[i]));

}

//сумма зарплаты сотрудника за 6 месяцев

if (str.Length < 7)

{

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

{

summMounth\_6 += int.Parse(str[i]);

}

}

else

{

for (int i = 0; i < 6; i++)

{

summMounth\_6 += int.Parse(str[i]);

}

}

//средняя зарплата сотрудника

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

{

avgSumm += int.Parse(str[i]);

}

avgSumm = avgSumm / str.Length;

}

public void Print() //метода вывода заполненной информации о сотруднике

{

Console.WriteLine("\n\tФамилия сотрудника: " + Surname);

for (int i = 0; i < Month\_salary.Count; i++)

Console.WriteLine($"\t\t\tМесяц: {Month\_salary.ElementAt(i).Key}\tЗарплата: {Month\_salary.ElementAt(i).Value}");

}

public string write\_File()

{

string file = $"Фамилия сотрудника: {Surname}\n\tЗарплата за 6 месяцев: {summMounth\_6}\n\tСредняя зарплата сотрудника: {avgSumm}";

return file;

}

}

class Program

{

static void Main(string[] args)

{

Worker worker1 = new Worker("Капская", random\_salary());

worker1.Print();

Worker worker2 = new Worker("Попов");

worker2.Print();

Worker worker3 = new Worker("Сибирев", random\_salary());

worker3.Print();

string random\_salary() //метод формирует зарплаты генератором случайных чисел и возвращает строкой, где каждая из зарплат перечисляется через запятую

{

Random rand = new Random();

int n = rand.Next(1, 11); //количество полученных сотрудником зарплат в диапазоне от 1 до 10

int[] mass\_salary = new int[n];

string str\_salary;

for (int i = 0; i < mass\_salary.Length; i++)

{

mass\_salary[i] = rand.Next(40000, 60001); //зарплата сотрудника в диапазоне от 40 000 до 60 000

}

str\_salary = mass\_salary.Select(a => a.ToString()).Aggregate((i, j) => i + "," + j); //преобразование массива зарплат в строку с разделителем ввиде запятой

return str\_salary;

}

//запись в файл фамилии сотрудников, зарплату за 6 месяцев и среднюю зарплату сотрудников

using (StreamWriter writer = new StreamWriter("File.txt"))

{

writer.WriteLine(worker1.write\_File());

writer.WriteLine(worker2.write\_File());

writer.WriteLine(worker3.write\_File());

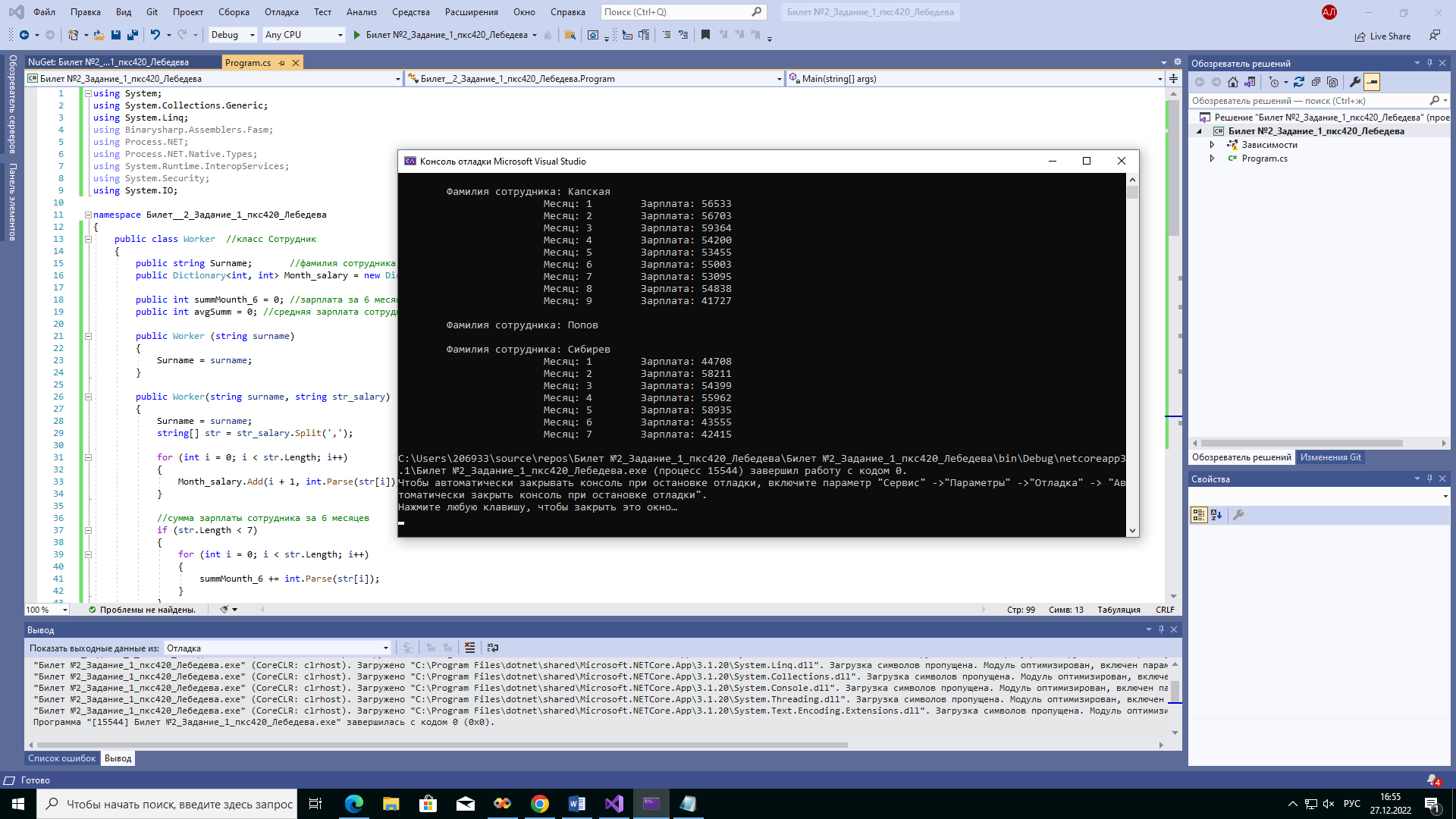
}

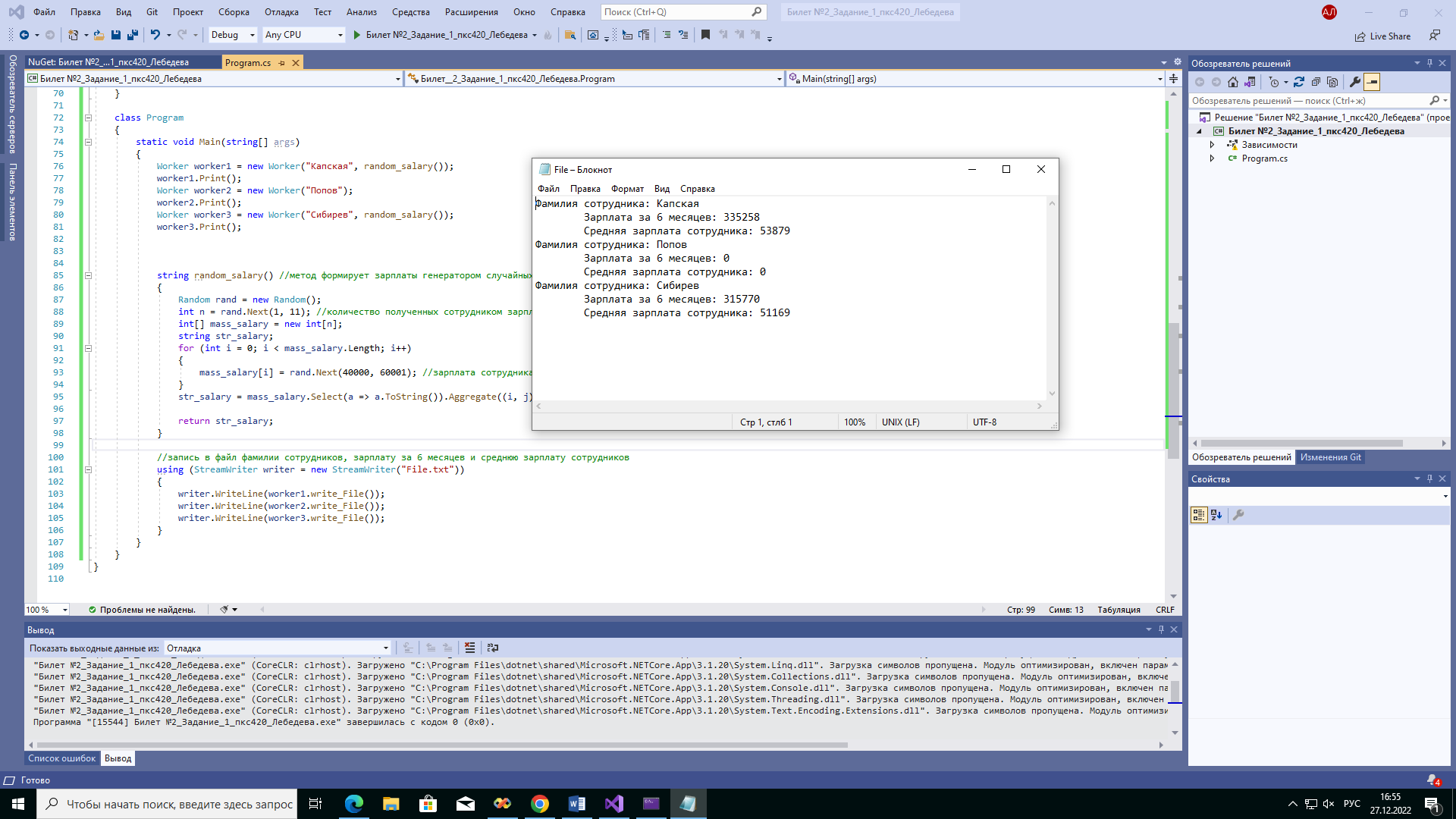
}

}

}

**Результат выполнения программы:**





**Ассемблерная вставка:**

using System;

using System.Collections.Generic;

using System.Linq;

using Binarysharp.Assemblers.Fasm;

using Process.NET;

using Process.NET.Native.Types;

using System.Runtime.InteropServices;

using System.Security;

using System.IO;

namespace ConsoleApp1

{

public class Worker //класс Сотрудник

{

public string Surname; //фамилия сотрудника

public Dictionary<int, int> Month\_salary = new Dictionary<int, int>(); //коллекция зарплат по номеру месяца

//public List<System.Int32> LabS = new List<System.Int32>();

public int[] mass = new int[10];

public Worker(string surname)

{

Surname = surname;

}

public Worker(string surname, int[] str\_salary)

{

mass = str\_salary;

Surname = surname;

int[] str = str\_salary;

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

{

Month\_salary.Add(i + 1, str[i]);

}

}

public void Print() //метода вывода заполненной информации о сотруднике

{

Console.WriteLine("\n\tФамилия сотрудника: " + Surname);

for (int i = 0; i < Month\_salary.Count; i++)

Console.WriteLine($"\t\t\tМесяц: {Month\_salary.ElementAt(i).Key}\tЗарплата: {Month\_salary.ElementAt(i).Value}");

}

public int Summ()

{

int a = 0;

;

if (!false)

{

FasmNet \_FasmNet = new FasmNet();

\_FasmNet.AddLine("use32");

\_FasmNet.AddLine("xor eax, eax");

\_FasmNet.AddLine("xor ebx, ebx");

\_FasmNet.AddLine("xor ecx, ecx");

\_FasmNet.AddLine("mov ebx, dword [ebp+12]");

\_FasmNet.AddLine("sum:");

\_FasmNet.AddLine("add eax, dword [ebx+ecx\*4]");

\_FasmNet.AddLine("add ecx, 1");

\_FasmNet.AddLine("cmp ecx, dword [ebp+8]");

\_FasmNet.AddLine("jne sum");

\_FasmNet.AddLine("ret");

System.Byte[] \_ByteS = \_FasmNet.Assemble();

Process.NET.Memory.IAllocatedMemory \_IAllocatedMemory =

new ProcessSharp(System.Diagnostics.Process.GetCurrentProcess(), Process.NET.Memory.MemoryType.Local)

.MemoryFactory

.Allocate(

name: "MyName",

size: \_ByteS.Length,

protection: MemoryProtectionFlags.ExecuteReadWrite

);

\_IAllocatedMemory.Write(0, \_ByteS);

delAssemb\_sum del\_sum = Marshal.GetDelegateForFunctionPointer<delAssemb\_sum>(\_IAllocatedMemory.BaseAddress);

a = del\_sum(mass.Length - 1, mass);

\_IAllocatedMemory.Dispose();

;

}

return a;

}

[System.Security.SuppressUnmanagedCodeSecurityAttribute]

[System.Runtime.InteropServices.UnmanagedFunctionPointerAttribute(System.Runtime.InteropServices.CallingConvention.Cdecl)]

public delegate int delAssemb\_sum(int \_count, params int[] \_ints);

public int Summ(int \_count, params int[] \_ints)

{

;

int qwds = \_ints[1];

int \_summ = 0;

int i = 0;

while (i < \_count)

{

int a = \_ints[i];

\_summ = \_summ + a;

i++;

}

return \_summ;

}

public int Middle()

{

int a = 0;

;

if (!false)

{

FasmNet \_FasmNet = new FasmNet();

\_FasmNet.AddLine("use32");

\_FasmNet.AddLine("xor eax, eax");

\_FasmNet.AddLine("xor ebx, ebx");

\_FasmNet.AddLine("xor ecx, ecx");

\_FasmNet.AddLine("mov ebx, dword [ebp+12]");

\_FasmNet.AddLine("sum:");

\_FasmNet.AddLine("add eax, dword [ebx+ecx\*4]");

\_FasmNet.AddLine("add ecx, 1");

\_FasmNet.AddLine("cmp ecx, dword [ebp+8]");

\_FasmNet.AddLine("jne sum");

\_FasmNet.AddLine("ret");

System.Byte[] \_ByteS = \_FasmNet.Assemble();

Process.NET.Memory.IAllocatedMemory \_IAllocatedMemory =

new ProcessSharp(System.Diagnostics.Process.GetCurrentProcess(), Process.NET.Memory.MemoryType.Local)

.MemoryFactory

.Allocate(

name: "MyName",

size: \_ByteS.Length,

protection: MemoryProtectionFlags.ExecuteReadWrite

);

\_IAllocatedMemory.Write(0, \_ByteS);

delAssemb\_sum del\_sum = Marshal.GetDelegateForFunctionPointer<delAssemb\_sum>(\_IAllocatedMemory.BaseAddress);

a = del\_sum(mass.Length - 1, mass);

\_IAllocatedMemory.Dispose();

;

}

return a/mass.Length;

}

}

public class Program

{

static void Main(string[] args)

{

Worker worker1 = new Worker("Капская", random\_salary());

worker1.Print();

Worker worker2 = new Worker("Попов");

worker2.Print();

Worker worker3 = new Worker("Альшакова", random\_salary());

worker3.Print();

Console.WriteLine("\nОбщая сумма зарплаты Капской: ");

System.Console.WriteLine(worker1.Summ());

Console.WriteLine("Средняя сумма зарплаты Капской: ");

System.Console.WriteLine(worker1.Middle());

Console.WriteLine("\nОбщая сумма зарплаты Попова: ");

System.Console.WriteLine(worker2.Summ());

Console.WriteLine("Средняя сумма зарплаты Попова: ");

System.Console.WriteLine(worker2.Middle());

Console.WriteLine("\nОбщая сумма зарплаты Альшаковой: ");

System.Console.WriteLine(worker3.Summ());

Console.WriteLine("Средняя сумма зарплаты Альшаковой: ");

System.Console.WriteLine(worker3.Middle());

System.Console.ReadLine();

int[] random\_salary() //метод формирует зарплаты генератором случайных чисел

{

Random rand = new Random();

int n = 10;

int[] mass\_salary = new int[n];

for (int i = 0; i < mass\_salary.Length; i++)

{

mass\_salary[i] = rand.Next(40000, 60001); //зарплата сотрудника в диапазоне от 40 000 до 60 000

}

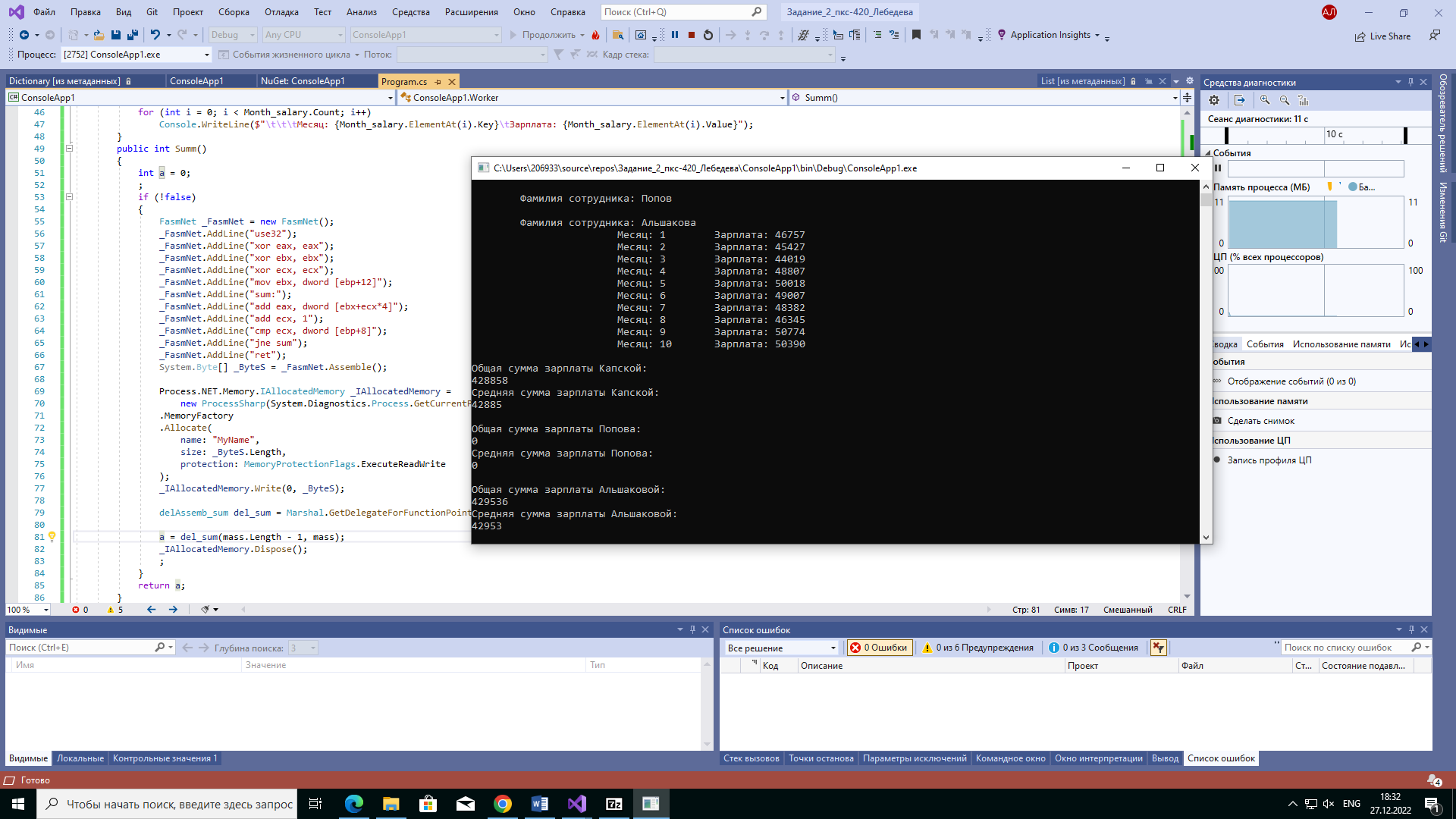
return mass\_salary;

}

}

}

}



**Вывод:** Я освоила работу с классами и коллекциями на языке С#, освоила вставку языка Assembler для обработки данных.