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-320

*М.А. Щусь*

*(И.О. Фамилия)*

Проверили:

Г.И. Киреева

*(И.О. Фамилия)*

И.В.Сибирев

*(И.О. Фамилия)*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Москва**

**2022**

**Вариант 2**

**Коды проекта С#:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.IO;

using System.Text;

namespace ConsoleApp1

{

class Worker // Создаем класс сотрудника

{

string surname; //Фамилия

List<int> paym = new List<int>(); //Коллекция выплат сотруднику

int fullpay; //Поле для суммы выплат

double mid; //Поле для средней выплаты

public void output() //Функция вывода в консоль информации о сотруднике

{

fullpay = paym.Sum(); //Расчет Суммы выплат

mid = Math.Round(paym.Average(), 2); //Расчет средней выплаты

Console.WriteLine("---------------------------------------------------");

Console.WriteLine("Фамилия работника: " + surname);

Console.WriteLine("Зарплаты: ");

for (int i = 0; i < paym.Count(); i++)

{

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

}

Console.WriteLine();

Console.WriteLine("Сумма зарплат: " + fullpay);

Console.WriteLine("Средняя зарплата: " + mid);

Console.WriteLine("---------------------------------------------------");

Console.WriteLine();

}

public void writefile() //Функция записи информации о сотруднике в файл

{

using (FileStream F = new FileStream("info.txt", FileMode.OpenOrCreate)) //Очистка потока и работа с файлом

{

F.Seek(0, SeekOrigin.End);

string strok = surname + " Средняя: " + mid + " Зарплаты за 6 месяцев: ";

if (paym.Count <= 6)

{

foreach (int o in paym)

{

strok += Convert.ToString(o) + " ";

}

}

else

{

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

{

strok += Convert.ToString(paym[i]) + " ";

}

}

strok += "\n";

byte[] input = Encoding.Default.GetBytes(strok);

F.Write(input, 0, input.Length);

}

}

public Worker(string a) //Конструктор с фамилией

{

surname = a;

}

public Worker(string a, string b) //Конструктор с фамилией и зарплатами

{

surname = a;

string[] money = b.Split(new char[] { ',' }, StringSplitOptions.RemoveEmptyEntries);

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

{

int xx = Convert.ToInt32(money[i]);

paym.Add(xx);

}

}

}

class Program

{

static void Main(string[] args)

{

using (FileStream A = new FileStream("info.txt", FileMode.OpenOrCreate)) //Очистка потока и файла после предыдущего запуска

{

A.SetLength(0);

}

List<Worker> workers = new List<Worker>(); //Создание коллекции с сотрудниками

try

{

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

{

Console.Write("Введите фамилию работника: ");

string a = Console.ReadLine();

Random rnd = new Random();

int contm = rnd.Next(1, 11);

Console.WriteLine("Количество выплат: " + contm);

string mks = "";

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

{

Random rn = new Random();

int x = rn.Next(40000, 60001);

mks += x + ",";

}

Worker A = new Worker(a, mks);

workers.Add(A);

}

for (int i = 0; i < workers.Count; i++)

{

workers[i].output();

workers[i].writefile();

}

}

catch (OverflowException) //Обработка исключений

{

Console.WriteLine("Переполнение!");

}

catch (FormatException)

{

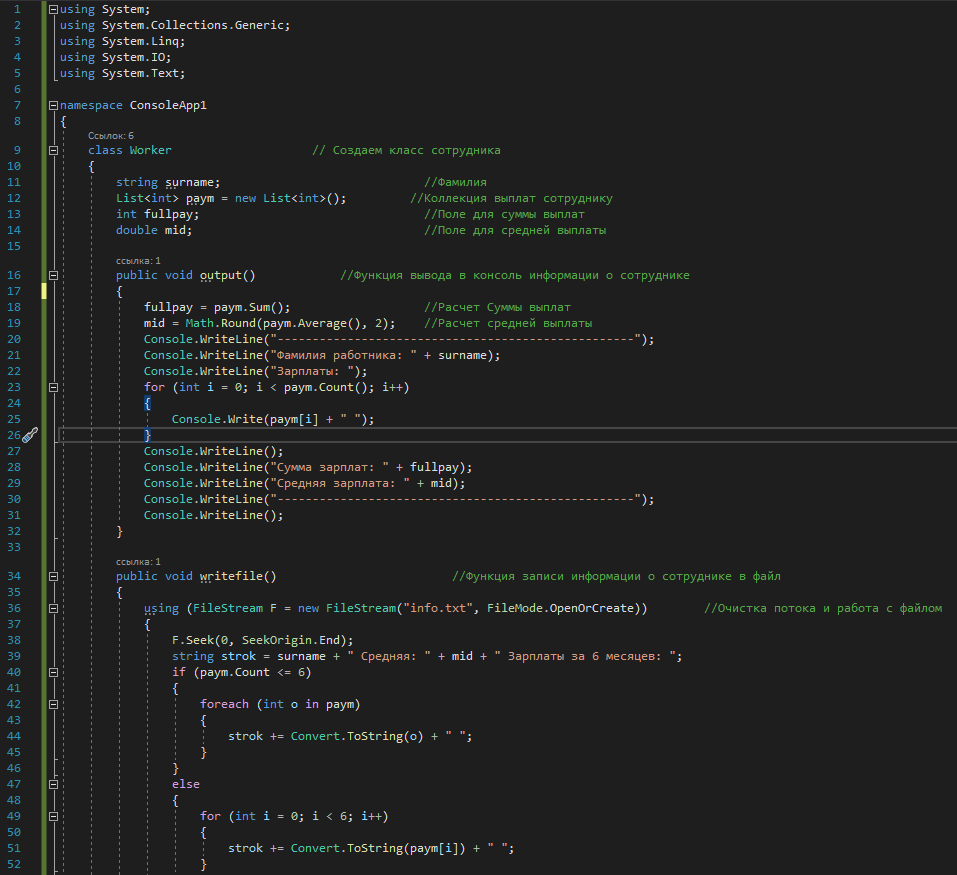
Console.WriteLine("Неверный формат!");

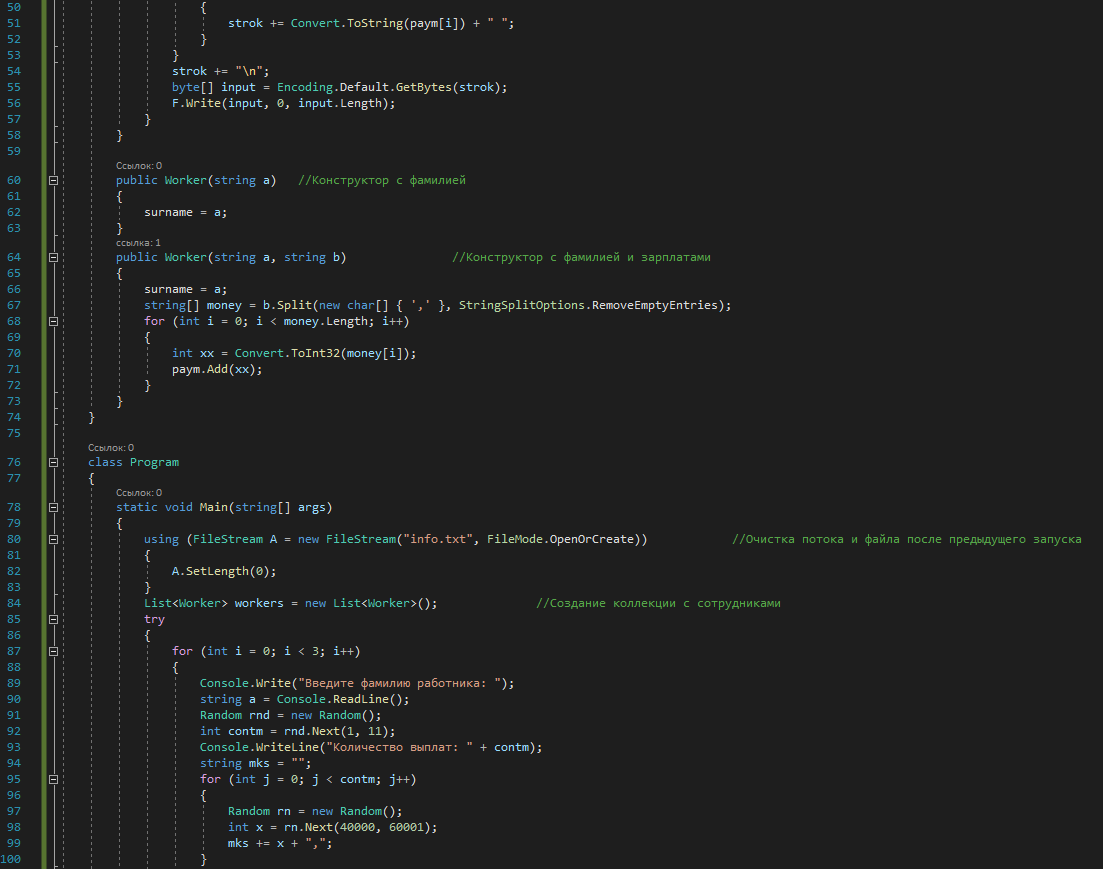
}

}

}

}

****

****

****

**Коды библиотеки C++ с ассемблерной вставкой:**

#include <iostream>

int SumAll(int a, int b[])

{

\_\_asm {

xor eax, eax

xor ebx, ebx

xor ecx, ecx

mov ebx, dword ptr [ebp + 8+4]

metka\_ForStart:

add eax, dword ptr [ebx]

add ebx, 4

add ecx, 1

cmp ecx, dword ptr [ebp + 8]

je metka\_Forend

jmp metka\_ForStart

metka\_Forend:

//return eax

}

}

int main()

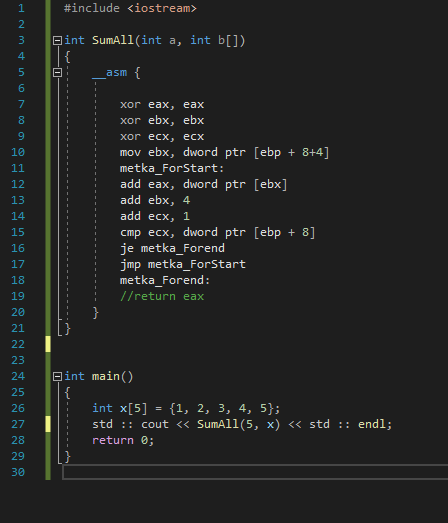
{

int x[5] = {1, 2, 3, 4, 5};

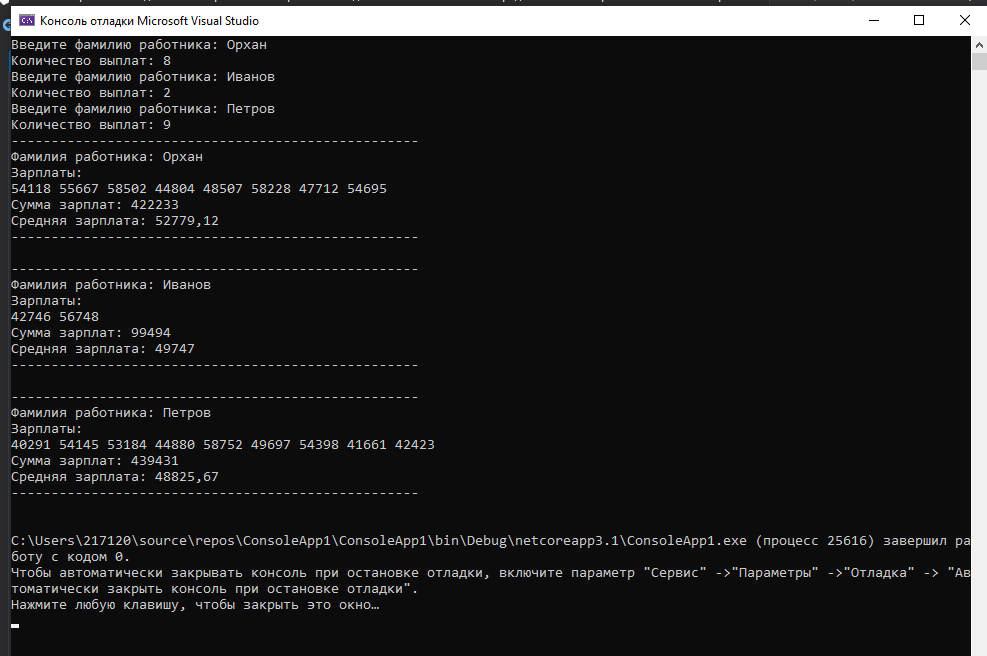
std :: cout << SumAll(5, x) << std :: endl;

return 0;

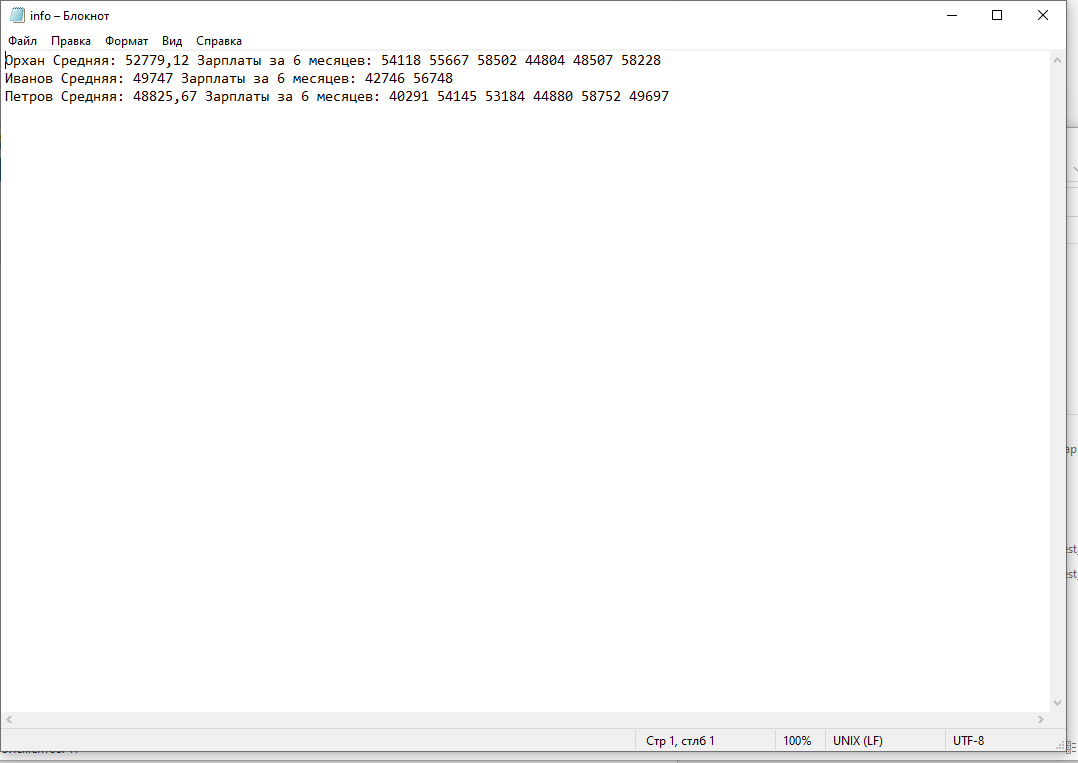
}

****

**Результат вывода в консоль:**

****

**Результат записи в текстовый файл:**

****