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

*\_\_\_Гнеушев Ярослав Романович\_\_*

*(И.О. Фамилия)*

Проверили:

Г.И. Киреева

*(И.О. Фамилия)*

И.В.Сибирев

*(И.О. Фамилия)*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Москва**

**2022**

Билет №1

Листинг программы:

#include <Windows.h>

#include <iostream>

#include <string>

#include <vector>

#include <fstream>

using namespace std;

class Student

{

string surname;

int marks[10];

int cnt = 0;

public:

Student();

Student(string \_surname)

{

surname = \_surname;

}

Student(string \_surname, string \_marks)

{

surname = \_surname;

cnt = 0;

for (int i = 0; i < \_marks.size(); i++)

{

if (\_marks[i] != ',')

{

marks[cnt++] = \_marks[i]-'0';

}

}

}

void displayMarks()

{

cout << "-------------------------\n";

cout << "Оценки студента " + surname << endl;

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

{

cout << marks[i] << " ";

}

cout << "\n-------------------------\n";

}

int\* getMarks()

{

return marks;

}

};

// Функция на случайное сичло в диапазоне (min,max)

int getRandomNum(int min, int max)

{

int range = max - min + 1;

srand((int)time(0));

return rand() % range + min;

}

// Ассемблер на получение суммы массива

int getMarksSum(int marks[], int len)

{

int res = 0;

\_\_asm

{

mov ecx, len

mov esi, marks

cycl :

mov eax, [esi]

push ecx

add res, eax

pop ecx

add esi, 4

loop cycl

}

return res;

}

// Ассемблер на % качества знаний

int studentsPerformance(int fours, int fives)

{

int res = 0;

\_\_asm

{

mov eax, fours

add eax, fives

mov ecx, 100

mul ecx

mov res, eax

}

return res;

}

// Вывод данных в файл

void printToFile(string surname, int nums[], int amount, float avgMark)

{

ofstream myfile("example.txt", ios\_base::app);

myfile << surname;

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

{

myfile << " " << nums[i];

}

myfile << " Средний балл: " << avgMark << endl;

myfile.close();

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

vector<Student> students;

string surname, marks;

string amount;

int fours = 0;

int fives = 0;

int totalMarks = 0;

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

{

int marksCount = getRandomNum(1, 10);

cout << "Введите фамилию студента №" << i + 1 << ": "; cin >> surname;

marks = "";

totalMarks += marksCount;

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

{

int mark = getRandomNum(2, 5);

marks += to\_string(mark) + ',';

if (mark == 4)

{

fours++;

}

if (mark == 5)

{

fives++;

}

Sleep(100);

}

Student s = Student(surname, marks);

students.push\_back(s);

s.displayMarks();

float avgMark = getMarksSum(s.getMarks(), marksCount) / (float)marksCount;

avgMark = floor(avgMark \* 100.0) / 100.0;

printToFile(surname, s.getMarks(), min(5,marksCount), avgMark);

cout << "Средний балл: " << avgMark << endl << endl;

}

int perf = studentsPerformance(fours, fives)/(float)totalMarks;

perf = floor(perf \* 100) / 100;

ofstream myfile("example.txt", ios\_base::app);

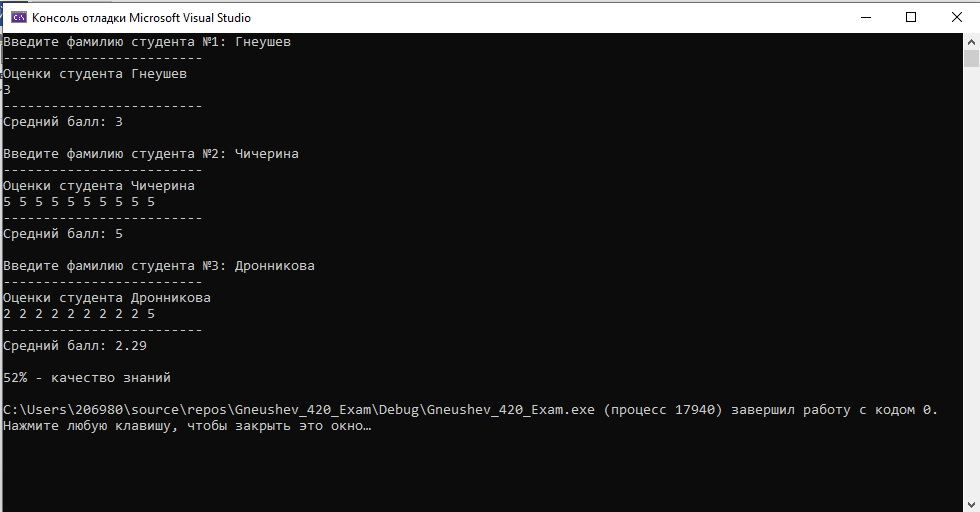
cout << perf << "% - качество знаний" << endl;

myfile << perf << "% - качество знаний" << endl;

myfile.close();

}

Вывод в консоль:



Вывод в файл:

