Выполнил

студент группы КТбо1-2 С.С. Абрамов

Принял

доцент кафедры САиТ В. С. Лапшин

Таганрог 2024

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

(ФГАОУ ВО «ЮФУ»)

Институт компьютерных технологий и информационной безопасности

Кафедра системного анализа и телекоммуникаций

ЛАБОРАТОРНАЯ РАБОТА №4

по дисциплине «Основы алгоритмизации и программирования»

**«РАЗРАБОТКА ПРИЛОЖЕНИЙ НА С++   
В ОБЪЕКТНОМ СТИЛЕ»**

Вариант 1

# Содержание

Содержание 2

Техническое задание 3

Цель задания 3

Задача 3

Ход работы 4

Написание программы 4

Пример работы программы 4

Вывод 5

Листинг 6

app.cpp 6

datamanager.h 7

datamanager.cpp 9

# Техническое задание

## Цель задания

Цель задания: ознакомиться с возможностью создания классов в С++, которые можно в последствии использовать для разработки приложения под Windows (CLI-приложение).

## Задача

Задачи лабораторной работы:

1. Разработать объектную модель приложения.
2. Создать CLI-консольное приложение.
3. Создать в среде VS классы разработанной объектной модели.
4. Отладить методы классов.

Заданием моего варианта является работа со списком оценок студентов и последующий вывод в файл списка студентов, имеющих определённое количество пятёрок, четвёрок, троек и двоек.

# Ход работы

## Написание программы

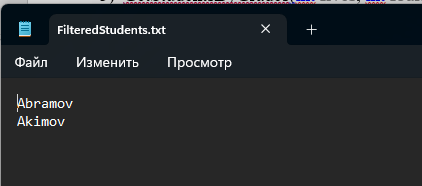
Для работы с тремя файлами я сделал свой класс **FileHandler**. Рассмотрю его поля и методы.

Класс имеет private поля **String^ TextFileName**. Они содержат адреса соответствующих текстовых файлов. Также в объявлен словарь **studentGrades,** в котором буду хранитьсястудентов и их оценок.

Рассмотрим public методы класса:

1. FileHandler(void) – инициализирует файл, ставя ему имя по умолчанию, и создание пустого словаря студентов и их оценок.
2. SetTextFileName(String^ NewName) – устанавливает новое имя для файла.
3. AddStudent(String^ name, List<int>^ grades) – добавляет студента в словарь.
4. DelStudent(String^ name) – удаление студента и информации о нём из словаря.
5. GetStudentsByGrades(int fives, int fours, int threes, int twos) – получение студентов по оценкам.
6. SaveToFileStudentsByGrades(int fives, int fours, int threes, int twos) – сохранение студентов в файл

## Пример работы программы



# Вывод

Итогом работы стал класс, который записывает в файл отфильтрованные имена учеников по заданным значениям.

# Листинг

## Alab\_4.cpp

#include "pch.h"

#include "FileHandler.h"

using namespace System;

using namespace System::Collections;

using namespace System::Collections::Generic;

int main(array<System::String^>^ args) – функция программы, принимающая массив строковых аргументов.

{

FileHandler^ fh = gcnew FileHandler(); - Создаем объект FileHandler

List<int>^ gradesA = gcnew List<int>(); - создаём новый список оценок

gradesA->Add(5);

gradesA->Add(5);

gradesA->Add(4);

gradesA->Add(3);

fh->AddStudent("Abramov", gradesA); - добавляем студента с его оценками

List<int>^ gradesK = gcnew List<int>(); - создаём новый список оценок

gradesK->Add(5);

gradesK->Add(5);

gradesK->Add(4);

gradesK->Add(2);

fh->AddStudent("Karasenko", gradesK); - - добавляем студента с его оценками

List<int>^ gradesKu = gcnew List<int>(); - создаём новый список оценок

gradesKu->Add(5);

gradesKu->Add(4);

gradesKu->Add(3);

gradesKu->Add(3);

fh->AddStudent("Kubanov", gradesKu); - - добавляем студента с его оценками

List<int>^ gradesAk = gcnew List<int>(); - создаём новый список оценок

gradesAk->Add(5);

gradesAk->Add(5);

gradesAk->Add(4);

gradesAk->Add(3);

fh->AddStudent("Akimov", gradesAk); - - добавляем студента с его оценками

fh->SetTextFileName("FilteredStudents.txt"); - Устанавливаем новое имя файла для сохранения

fh->SaveToFileStudentsByGrades(2, 1, 1, 0); - Получаем и сохраняем студентов с двумя "5", одной "4" и одной "3"

return 0;

}

## FileHandler.h

using namespace System;

using namespace System::Collections;

using namespace System::Collections::Generic;

ref class FileHandler

{

public:

// Конструктор класса

FileHandler(void);

// Добавить студента с оценками в список

void AddStudent(String^, List<int>^);

// Удалить студента из списка

void DelStudent(String^);

// Установить новое имя файла для сохранения результатов поиска

void SetTextFileName(String^);

// Получить список студентов с заданным количеством оценок

ArrayList^ GetStudentsByGrades(int, int, int, int);

// Сохранить список студентов с заданным количеством оценок

void SaveToFileStudentsByGrades(int, int, int, int);

private:

String^ TextFileName; - имя файла для сохранения

Dictionary<String^, List<int>^>^ studentGrades; - словарь для хранения студентов и их оценок

};

## FileHandler.cpp

#include "pch.h"

#include "FileHandler.h"

using namespace System;

using namespace System::Collections;

using namespace System::Collections::Generic;

FileHandler::FileHandler(void)

{

TextFileName = gcnew String("Students.txt"); - инициализация имени файла по умолчанию.

studentGrades = gcnew Dictionary<String^, List<int>^>(); - создание пустого словаря студентов и их оценок.

}

void FileHandler::SetTextFileName(String^ NewName) – функция для установки нового имени файла.

{

TextFileName = NewName;

}

void FileHandler::AddStudent(String^ name, List<int>^ grades) – функция для добавления студента.

{

studentGrades[name] = grades;

}

void FileHandler::DelStudent(String^ name) – функция для удаления студента.

{

studentGrades->Remove(name);

}

ArrayList^ FileHandler::GetStudentsByGrades(int fives, int fours, int threes, int twos) – функция для получения студентов по их оценкам.

{

ArrayList^ result = gcnew ArrayList(); - создание нового пустого списка.

for each (KeyValuePair<String^, List<int>^> student in studentGrades) – перебор всех студентов в словаре.

{

int countFives = 0;

int countFours = 0;

int countThrees = 0;

int countTwos = 0;

for each (int grade in student.Value) – перебор всех оценок студента.

{

if (grade == 5) countFives++;

else if (grade == 4) countFours++;

else if (grade == 3) countThrees++;

else if (grade == 2) countTwos++;

}

if (countFives == fives && countFours == fours && countThrees == threes && countTwos == twos)

{

result->Add(student.Key); - добавление студента в созданный в функции список.

}

}

return result;

}

void FileHandler::SaveToFileStudentsByGrades(int fives, int fours, int threes, int twos) – функция для сохранения студента в файл.

{

ArrayList^ students = GetStudentsByGrades(fives, fours, threes, twos);

for each (String ^ student in students) – перебор всех студентов из полученного списка.

{

System::Console::WriteLine(student); - вывод имени студента в консоль.

}

System::IO::StreamWriter^ sw = gcnew System::IO::StreamWriter(this->TextFileName);

for each (String ^ student in students)

{

sw->WriteLine(student); - запись студента в файл.

}

sw->Close();

}