Министерство образования и науки Российской Федерации  
Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский  
политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЁТ**

**по лабораторной работе №18.10**

Дисциплина: «Информатика»

Тема: Сохранение данных в файле с использованием потоков

Вариант 7

Выполнил:

Студент группы РИС-20-1б

Нечаев Д.А.

Проверила:

Доцент кафедры ИТАС

Полякова О. А.

**Пермь, 2021**

**Цель задачи**

1) Создание консольного приложения, состоящего из нескольких файлов в системе программирования Visual Studio.

2) Разработка программы, в которой данные сохраняются в файле, корректируются и выводятся из файла на печать. Работа с файлом осуществляется с использованием потоковых классов.

Для достижения поставленной цели необходимо решить следующие задачи:

* Проанализировать теоретические сведения о ООП
* Абстрактные типы данных. Контейнеры
* Реализовать алгоритм поставленной задачи
* Учесть все исключительные ситуации на ввод элементов
* Создать отдельные файлы: .cpp для описания методов класса, .cpp для описания главной функции, .h для описания класса

Постановка задачи

Создать класс Money для работы с денежными суммами. Число должно быть представлено двумя полями: типа long для рублей и типа int для копеек. Дробная часть числа при выводе на экран должна быть отделена от целой части запятой

Задание:

• Удалить все записи не равные заданному значению.

• Уменьшить все записи с заданным значением в два раза.

• Добавить K записей в начало файла.

Анализ задачи

1. Определить какие операции должны быть выполнены по заданию:

* Создание класса

class Money

* Создание группы сумм, заполнение класса
* Создание методов получения суммы по номеру, печати, конструкторов
* Отчистка памяти

1. Для решения задачи используются переменные:

Несколько переменных целочисленного значения для ввода чисел и реализации меню

int s, in, menu = 4;

Указатель для работы с динамической памятью

1. Ввод данных осуществляется посредством функции

cin >> rub;

Вывод данных осуществляется посредством функции

cout << "Искомый элемент = " << a.give(in);

1. Создание метода класса , который печатает сумму:

void Money::Print()

{

cout << "\nРубли - " << rub << ", копейки - " << cop;

1. Создание методов класса для нахождения размера и доступа по индексу

int give(int index);

int givesize();

Создание контейнера

Money::Money(int r, long int c)

{

rub = r;

cop = c;

double co = c;

sum = r + co / 100;

}

1. Создание инструмента обработки исключительных ситуаций. Для этого была использована логика try:

try

{

cout << "Искомый элемент: " << a.give(in);

}

catch (const int in)

{

cout << "Ошибка ввода";

}

}

Код

Main.cpp:

#include <iostream>

#include "MONEY.h"

#include <windows.h>

#include <sstream>

#include <fstream>

#include <string>;

using namespace std;

int main()

{

string path = "FILE.txt";

ofstream fout;

fout.open(path);

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(LC\_ALL, "Russian");

int s = 0, r = 0, menu = 1, index, k = 0;

double zn;

long int c = 0;

while (s <= 0)

{

cout << "Введи количество сумм: "; cin >> s;

}

Money\* data;

data = new Money[s];

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

{

cout << "Введи количество рублей: "; cin >> r;

cout << "Введи количество копеек: "; cin >> c; cout << endl;

data[i] = Money(r, c);

fout << data[i].Get() << " ";

fout.close();

}

while (menu == 1 || menu == 2 || menu == 3 || menu == 4 || menu == 5)

{

cout << endl << endl << "Меню: \n\t1. Вывести все суммы в консоль \n\t2. Обновить файл \n\t3. Удалить все записи, равные значению \n\t4. Уменьшить все записи с заданным значением в два раза \n\t5. Добавить K записей в начало файла \n\n\t0. Выход";

cout << endl << "Выбери пункт меню: "; cin >> menu;

if (menu == 1)

{

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

{

cout << data[i].Get() << " ";

}

}

if (menu == 2)

{

fout.open(path);

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

fout << data[i].Get() << " ";

cout << "Файл обновлен";

fout.close();

}

if (menu == 3)

{

cout << "Введи значение: "; cin >> zn;

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

{

if (data[i].sum == zn)

{

data[i].sum = -1;

k++;

}

}

Money\* data1;

data1 = new Money[s - k];

int j = 0;

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

{

if (data[i].sum != -1)

{

data1[j] = data[i];

j++;

}

}

delete[] data;

data = data1;

s = s - k;

k = 0;

cout << "Элементы удалены";

}

if (menu == 4)

{

cout << "Введи значение: "; cin >> zn;

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

{

if (data[i].sum == zn)

{

data[i].sum = data[i].sum / 2;

}

}

cout << "Изменения внесены";

}

if (menu == 5)

{

cout << "Сколько записей добавить: "; cin >> k;

Money\* data1;

data1 = new Money[s + k];

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

{

cout << "Введи количество рублей: "; cin >> r;

cout << "Введи количество копеек: "; cin >> c; cout << endl;

data1[i] = Money(r, c);

}

int j = 0;

for (int i = k; i < s + k; i++)

{

data1[i] = data[j];

j++;

}

delete[] data;

data = data1;

s = s + k;

k = 0;

cout << "Изменения внесены";

}

}

fout.close();

}

MHETODS.cpp:

#include <iostream>

#include <sstream>

#include <fstream>

using namespace std;

#include "MONEY.h"

Money::Money(int r, long int c)

{

rub = r;

cop = c;

double co = c;

sum = r + co / 100;

}

double Money::Get()

{

return sum;

}

void Money::Print()

{

cout << "\nРубли - " << rub << ", копейки - " << cop;

}

MONEY.h:

#include <iostream>

#include <sstream>

#include <fstream>

using namespace std;

class Money

{

public:

long int rub;

int cop;

double sum;

Money()

{

rub = cop = sum = 0;

}

Money(int r, long int c);

double Get();

void Print();

};

Блок-схема:

Main.cpp:



MHETODS.cpp:



MONEY.h:



Работа кода



