Институт Космических и Информационных технологий Сибирского Федерального Университета  
**Факультет Информатики и вычислительной техникиКафедра Вычислительной техники**

**Отчёт по лабораторной работе №15**

**«Основы ООП»**

Выполнил студент гр. КИ

Проверила:

Красноярск 2015

**Цели работы:**

Цель работы: освоение основных приемов обработки текстов с использованием стандартной библиотеки С++.

**Задание:**

Разработать программу, содержащую описание класса Money(деньги), который должен включать:

* Закрытые поля для хранения рублей и копеек.
* Конструктор (или несколько конструкторов) для создания объектов.
* Метод, демонстрирующий на экране денежную сумму в словесно-числовом формате. Например: 8 рублей 15 копеек.
* Метод, определяющий  количество копеек в заданной сумме

longKopeck ()

Требования к программе:

* В главной функции объявить 3 объекта класса Money.
* Выполнить инициализацию полей объектов различными способами.
* Вывести на экран в словесно-числовом формате денежные суммы, хранящиеся в каждом объекте.
* Рассчитать разницу  в копейках между денежными суммами, хранящимися в 1 и 2 объектах, и вывести ее на экран.

**Код программы:**

#include<iostream>

#include<clocale>

using namespace std;

class Money

{

private:

unsigned int r, k;

public:

// инициализирующийконструктор

Money(unsigned int R, unsigned int K)

{

if(K < 100&&R>0&&K>0)

r = R, k = K;

else

r = k = 0, cerr << "incorrect money format" << endl;

}

// конструктор по умолчанию

Money(){}

void PrintEn()

{

cout << "\n" << r << "рублей" << k << "копеек"<<endl;

}

unsigned int GetRub() { return r; }

unsigned int GetKop() { return k; }

void SetRub(unsigned int \_r) { if(\_r > 0) r =\_r; else r = k = 0, cerr << "incorrect rubles" << endl; }

void SetKop (unsigned int \_k) { if(\_k < 100 && \_k>0) k =\_k; else k = r = 0, cerr << "incorrect kopeiki" << endl; }

//разница в секундах с заданным временем

int Difference (Money m) { return (m.GetRub()- r)\*100 + (m.GetKop() -k);}

};

int main()

{

//Создаем, инициализируем и выводим первый объект

setlocale(LC\_ALL, "russian");

unsigned int r, k;

cout << "\n Enter:\n r= ";

cin >> r;

cout << " k= ";

cin >> k;

Money m1 (r, k);// используется инициализирующий конструктор

cout << "\nMoney 1 ";

m1.PrintEn ();

//Создаем, инициализируем и выводим второй объект

Money m2 (10, 15); // используется инициализирующий конструктор

cout << "\nMoney 2 ";

m2.PrintEn();

//Создаем, инициализируем и выводим третий объект

Money m3; // используется конструктор по умолчанию

m3.SetRub (14) ;

m3.SetKop (15) ;

cout << "\nMoney 3";

m3.PrintEn ();

// Определяем разницу в секундах между первым и вторым объектом

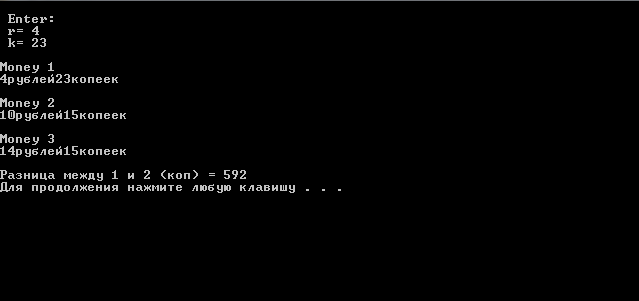
int kk = m1.Difference(m2);

cout << "\nРазница между 1 и 2 (коп) = " << kk << endl;

system("pause");

}

**Протокол отладки:**



**Вывод:** Были освоены приемы работы с динамическими двусвязными списками в c++.