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 №18.3**

Дисциплина: «Информатика»

Тема: Структуры данных

Вариант 7

Выполнил:

Студент группы РИС-20-1б

Нечаев Д.А.

Проверила:

Доцент кафедры ИТАС

Полякова О. А.

**Пермь, 2021**

**Цель задачи**

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

Использование перегруженных операций в классах

Для достижения поставленной цели необходимо решить следующие задачи:

* Проанализировать теоретические сведения о ООП
* Проанализировать понятия классов, инструменты для их создания
* Разобрать поля и методы классов
* Понять, как пользоваться перегруженными операциями в классах
* Реализовать алгоритм поставленной задачи
* 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. Реализовать

* Сложение денежный сумм
* Вычитание денежных сумм

Анализ задачи

1. Определить какие операции должны быть выполнены по заданию:

* Создание класса

class Money

* Создание Конструктора

Money::Money(long int rubs, int kops)

* Создание метода вывода информации класса
* Ввод данных

1. Для решения задачи используются переменные:

Несколько переменных целочисленного значения для ввода чисел

int rub = -1;

int kop = -1;

Поле типа double для хранения числа

double sum;

Переменные целочисленного типа для работы конструктора

long int rubs, int kops

1. Ввод данных осуществляется посредством функции

cin >> rub;

Вывод данных осуществляется посредством функции

cout << "Введи количество рублей: ";

1. Создание метода Print, который печатает данные конкретного контейнера

void Money::Print(double s)

{

cout << "Сумма = " << s << endl;

}

1. Создание контейнера

Money::Money(long int rubs, int kops)

{

double sum1 = kops;

sum1 = sum1 / 100;

sum1 = sum1 + rubs;

sum = sum1;

}

Код

Source.cpp:

#include "Money.h"

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

long int rub = -1;

int kop = -1;

while (rub < 1)

{

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

}

while (kop < 1 || kop >99)

{

cout << "Введи количество копеек: "; cin >> kop;

}

Money a(rub, kop);

a.Print(a.sum);

rub = -1; kop = -1;

while (rub < 1)

{

cout << endl << "Введи количество рублей: "; cin >> rub;

}

while (kop < 1 || kop >99)

{

cout << "Введи количество копеек: "; cin >> kop;

}

Money b(rub, kop);

b.Print(b.sum);

cout.precision(3);

cout << endl << "Деление первой суммы на вторую: " << a.sum / b.sum;

double k;

cout << endl << "Введи число, на которое будут умноженны суммы: "; cin >> k;

a.sum = a.sum \* k;

b.sum = b.sum \* k;

a.Print(a.sum);

b.Print(b.sum);

}

Money.cpp:

#include "Money.h"

#include <iostream>

using namespace std;

Money::Money(long int rubs, int kops)

{

double sum1 = kops;

sum1 = sum1 / 100;

sum1 = sum1 + rubs;

sum = sum1;

}

void Money::Print(double s)

{

cout << "Сумма = " << s << endl;

}

Money.h:

#include <iostream>

using namespace std;

class Money

{

public:

double sum;

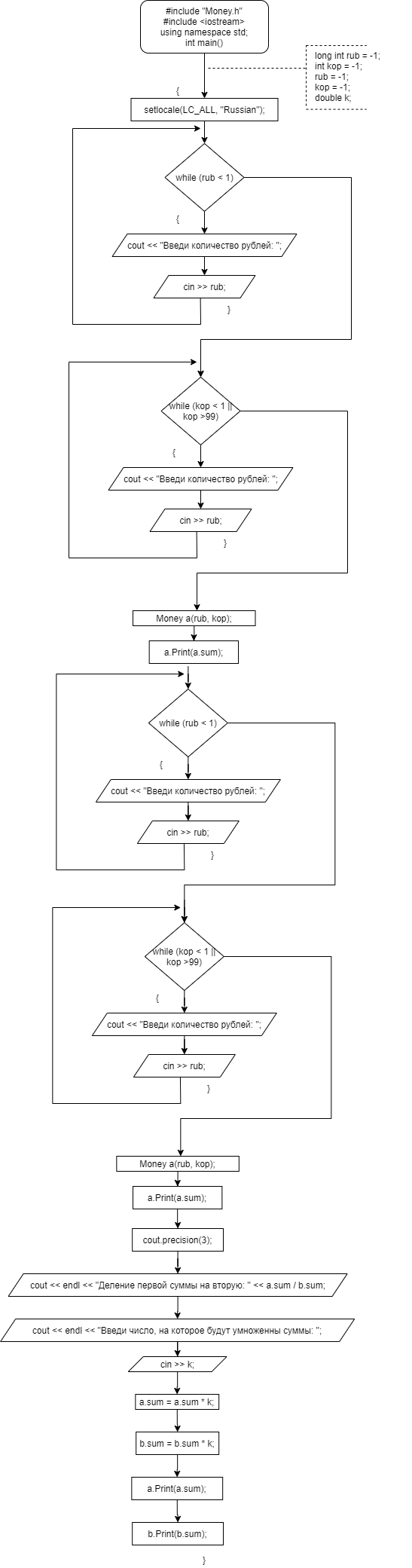
Money(long int rubs, int kops);

void Print(double s);

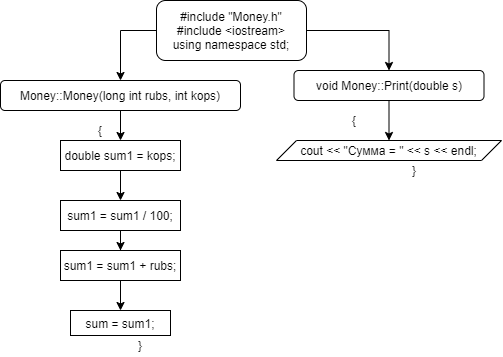
};

Блок-схема:

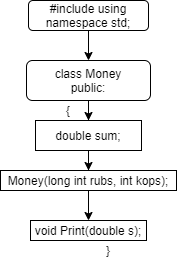
Source.cpp:



Money.cpp:



Money.h:



Работа кода

