

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
Пермский национальный исследовательский политехнический  
университет  
Электротехнический факультет  
Кафедра информационных технологий и автоматизированных систем

Лабораторная работа №2

**" Классы и объекты. Использование конструкторов"**

Вариант: 12

Выполнил студент ИВТ-24-26:  
Шишкин Максим Григорьевич

\_\_\_\_\_  
(дата, подпись)

Проверил доцент кафедры ИТАС:

Полякова Ольга Андреевна

\_\_\_\_\_  
(дата, подпись)

Пермь 2025

## Содержание

1 Постановка задачи .....	3
2 Код на C++ .....	4-5
3 Результаты работы .....	5
4 Блок-схема .....	6-8
5 Ответы на контрольные вопросы .....	9
6 Ссылка на github.....	9

## **1 Постановка задачи**

1. Определить пользовательский класс.
2. Определить в классе следующие конструкторы: без параметров, с параметрами, копирования.
3. Определить в классе деструктор.
4. Определить в классе компоненты-функции для просмотра и установки полей данных (селекторы и модификаторы).
5. Написать демонстрационную программу, в которой продемонстрировать все три случая вызова конструктора-копирования, вызов конструктора с параметрами и конструктора без параметров.

Вариант: Пользовательский класс КВИТАНЦИЯ, Номер – int, Дата – string, Сумма – double

## 2 Код на C++

```
main.cpp  ▢  Kvitanciya.cpp  Kvitanciya.h
2лабаперделка  (Глобальная область)

1  #include "Kvitanciya.h"
2  #include <iostream>
3  #include <locale>
4  #include <windows.h>
5
6  using namespace std;
7
8  int main() {
9      setlocale(LC_ALL, "ru");
10     SetConsoleCP(1251);
11     SetConsoleOutputCP(1251);
12
13     cout << "\n1. Создание kv1 (конструктор без параметров):" << endl;
14     Kvitanciya kv1;
15     kv1.display();
16
17     cout << "\n2. Создание kv2 (конструктор с параметрами):" << endl;
18     Kvitanciya kv2(123, "15.05.2023", 1500.75);
19     kv2.display();
20
21     cout << "\n3. Создание kv3 (конструктор копирования - явное копирование):" << endl;
22     Kvitanciya kv3 = kv2;
23     kv3.display();
24
25     cout << "\n4. Передача объекта в функцию (вызов конструктора копирования):" << endl;
26     Kvitanciya kv4(kv2);
27     kv4.display();
28
29     cout << "\n5. Возврат объекта из функции (вызов конструктора копирования):" << endl;
30     Kvitanciya kv5 = []() {
31         Kvitanciya temp(456, "20.05.2023", 2000.0);
32         return temp;
33     }();
34     kv5.display();
35
36     cout << "\n6. Работа с селекторами и модификаторами:" << endl;
37     kv1.setNumber(100);
38     kv1.setDate("10.05.2023");
39     kv1.setSumma(999.99);
40     cout << "Измененная kv1:" << endl;
41     cout << "Номер: " << kv1.getNumber() << endl;
42     cout << "Дата: " << kv1.getDate() << endl;
43     cout << "Сумма: " << kv1.getSumma() << endl;
44
45     cout << "\nКонец работы программы. Сейчас будут вызваны деструкторы." << endl;
46     return 0;
47 }
```

```
main.cpp  ▢  Kvitanciya.cpp  ▢  Kvitanciya.h
2лабаперделка  (Глобальная область)

1  #include "Kvitanciya.h"
2  #include <iostream>
3
4  using namespace std;
5
6  Kvitanciya::Kvitanciya() {
7      number = 0;
8      date = "01.01.2000";
9      summa = 0.0;
10     cout << "Вызван конструктор без параметров" << endl;
11 }
12
13 Kvitanciya::Kvitanciya(int num, string dt, double sm) {
14     number = num;
15     date = dt;
16     summa = sm;
17     cout << "Вызван конструктор с параметрами" << endl;
18 }
19
20 Kvitanciya::Kvitanciya(const Kvitanciya& other) {
21     number = other.number;
22     date = other.date;
23     summa = other.summa;
24     cout << "Вызван конструктор копирования" << endl;
25 }
26
27 Kvitanciya::~Kvitanciya() {
28     cout << "Вызван деструктор для квитанции №" << number << endl;
29 }
30
31 int Kvitanciya::getNumber() const { return number; }
32 string Kvitanciya::getDate() const { return date; }
33 double Kvitanciya::getSumma() const { return summa; }
34
35 void Kvitanciya::setNumber(int num) { number = num; }
36 void Kvitanciya::setDate(string dt) { date = dt; }
37 void Kvitanciya::setSumma(double sm) { summa = sm; }
38
39 void Kvitanciya::display() const {
40     cout << "Квитанция №" << number << endl;
41     cout << "Дата: " << date << endl;
42     cout << "Сумма: " << summa << " руб." << endl;
43 }
```



```
main.cpp  Kvitanciya.cpp  Kvitanciya.h  +  X
+ 2лабаперделка
1  #pragma once
2  #include <string>
3  using namespace std;
4
5  class Kvitanciya {
6  private:
7      int number;
8      string date;
9      double summa;
10
11 public:
12     Kvitanciya();
13     Kvitanciya(int num, string dt, double sm);
14     Kvitanciya(const Kvitanciya& other);
15     ~Kvitanciya();
16
17     int getNumber() const;
18     string getDate() const;
19     double getSumma() const;
20
21     void setNumber(int num);
22     void setDate(string dt);
23     void setSumma(double sm);
24
25     void display() const;
26 };
```

### 3 Результаты работы

```
1. Создание kv1 (конструктор без параметров):
Вызван конструктор без параметров
Квитанция %i0
Дата: 01.01.2000
Сумма: 0 руб.

2. Создание kv2 (конструктор с параметрами):
Вызван конструктор с параметрами
Квитанция %i123
Дата: 15.05.2023
Сумма: 1500.75 руб.

3. Создание kv3 (конструктор копирования – явное копирование):
Вызван конструктор копирования
Квитанция %i123
Дата: 15.05.2023
Сумма: 1500.75 руб.

4. Передача объекта в функцию (вызов конструктора копирования):
Вызван конструктор копирования
Квитанция %i123
Дата: 15.05.2023
Сумма: 1500.75 руб.

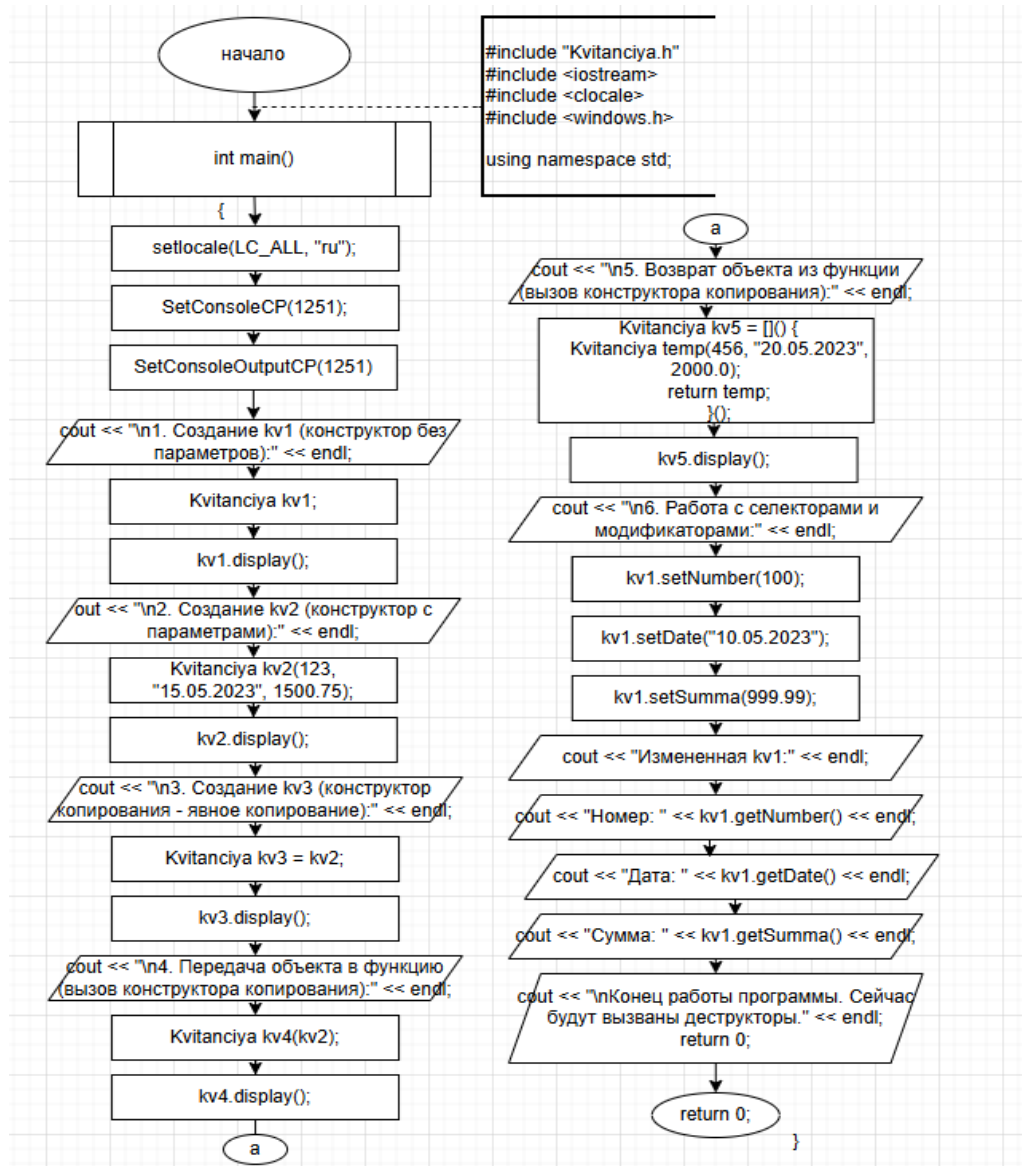
5. Возврат объекта из функции (вызов конструктора копирования):
Вызван конструктор с параметрами
Квитанция %i456
Дата: 20.05.2023
Сумма: 2000 руб.

6. Работа с селекторами и модификаторами:
Измененная kv1:
Номер: 100
Дата: 10.05.2023
Сумма: 999.99

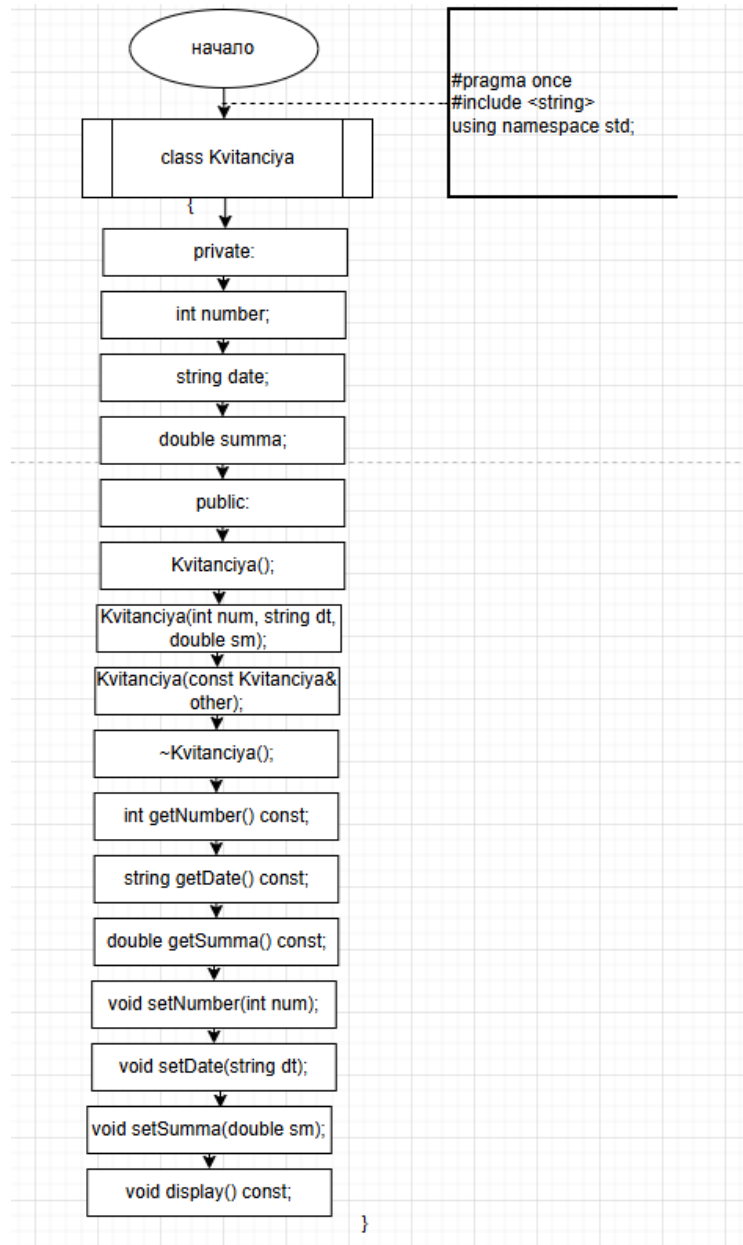
Конец работы программы. Сейчас будут вызваны деструкторы.
Вызван деструктор для квитанции %i456
Вызван деструктор для квитанции %i123
Вызван деструктор для квитанции %i123
Вызван деструктор для квитанции %i123
Вызван деструктор для квитанции %i100
```

## 4 Блок-схема

Main.cpp

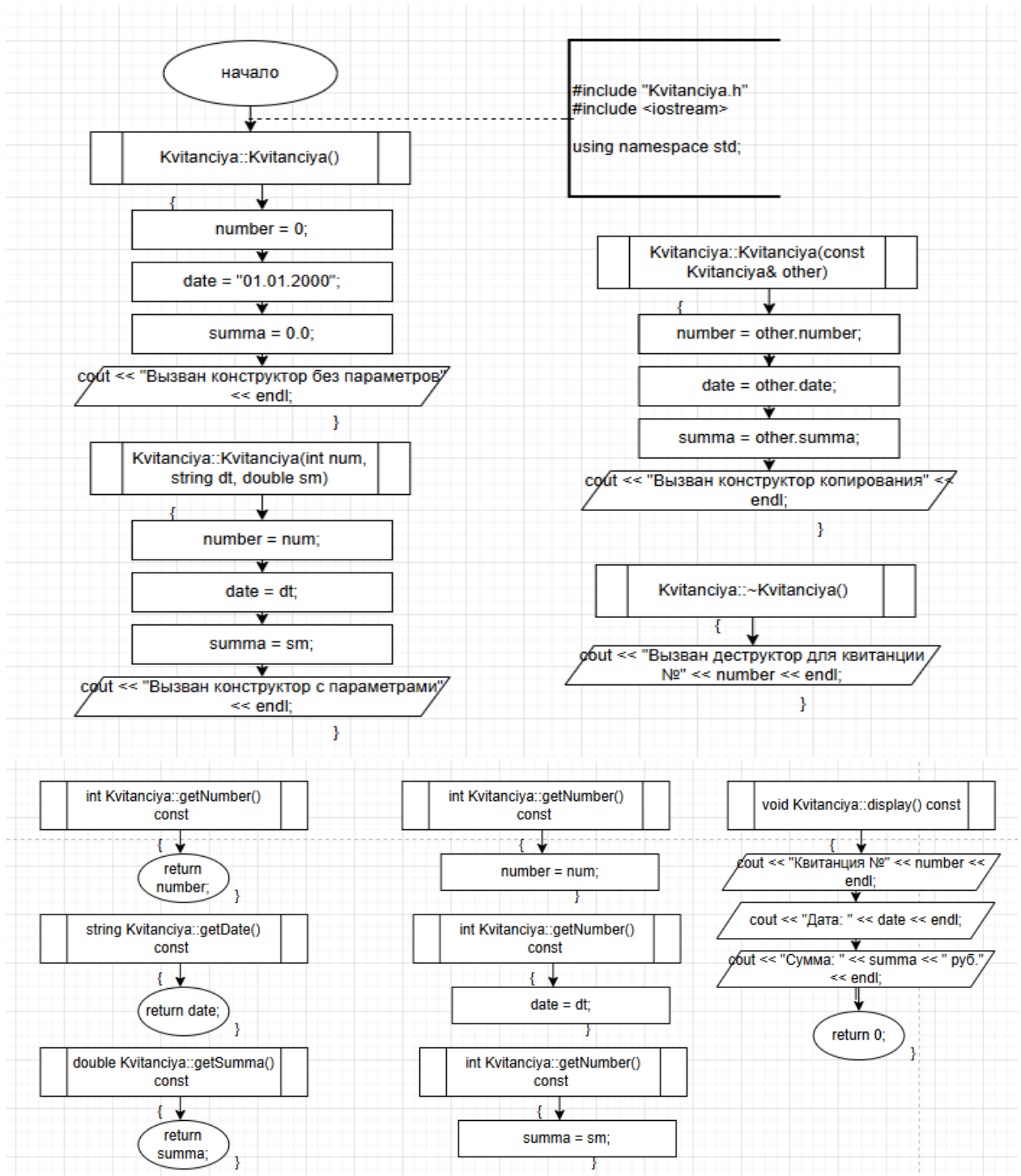


## Kvitanciya.h





# Kvitanciya.cpp



## 5 Ответы на контрольные вопросы

1. Для инициализации объектов при создании.
2. Три: без параметров, с параметрами, копирования.
3. Для освобождения ресурсов; явно — если нужно освободить память или закрыть файл.
4. Без параметров — для значений по умолчанию; с параметрами — для инициализации; копирования — для создания копии объекта.
5. При передаче по значению, возврате из функции, создании копии.
6. Имя как у класса, не возвращают значение, могут быть перегружены.
7. Имя с ~, не имеет параметров, вызывается один раз при уничтожении.
8. Ко всем — в том числе private.
9. Указатель на текущий объект.
10. Внутри — компактнее, вне — через ::, можно разделить интерфейс и реализацию.
11. Не возвращает значения.
12. Конструктор по умолчанию, копирования, деструктор, оператор присваивания.
13. Не возвращает значения.
14. Конструктор по умолчанию.
15. Конструктор по умолчанию.
16. Конструктор с параметрами.
17. Конструктор с параметрами и конструктор копирования.
18. Конструктор с параметрами и оператор присваивания.
19. Конструктор копирования.
20. Через метод: `p.set_name("НовоеИмя");`.

## 6 Ссылка на github

ссылка на github - <https://github.com/MAKSPOWERO/mas1>