Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Электротехнический факультет  
Кафедра «Информационные технологии и автоматизированные системы» направление подготовки: 09.03.04 – «Программная инженерия»

# Лабораторная работа " Классы и объекты. Использование конструкторов."

Выполнил студент гр. РИС-24-3б

Жиряков Леонид Антонович

Проверил:

Доцент кафедры ИТАС   
Ольга Андреевна Полякова

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (оценка) (подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

г. Пермь, 2024

Вариант 8.

Постановка задачи:

Общая:

1. Определить пользовательский класс.

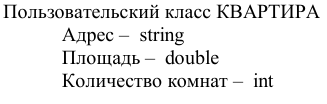
2. Определить в классе следующие конструкторы: без параметров, с параметрами, копирования.

3. Определить в классе деструктор.

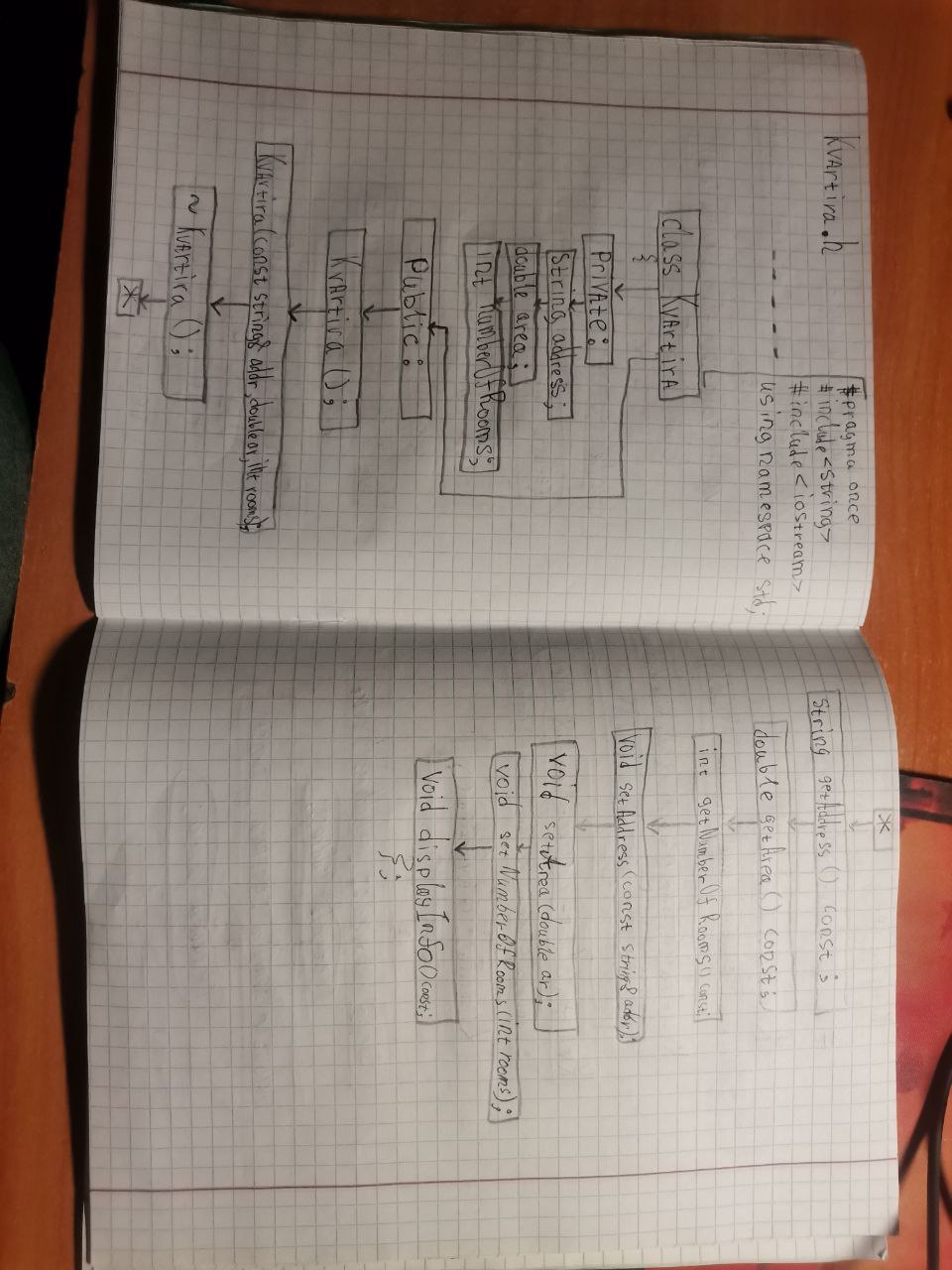
4. Определить в классе компоненты-функции для просмотра и установки полей, данных (селекторы и модификаторы).

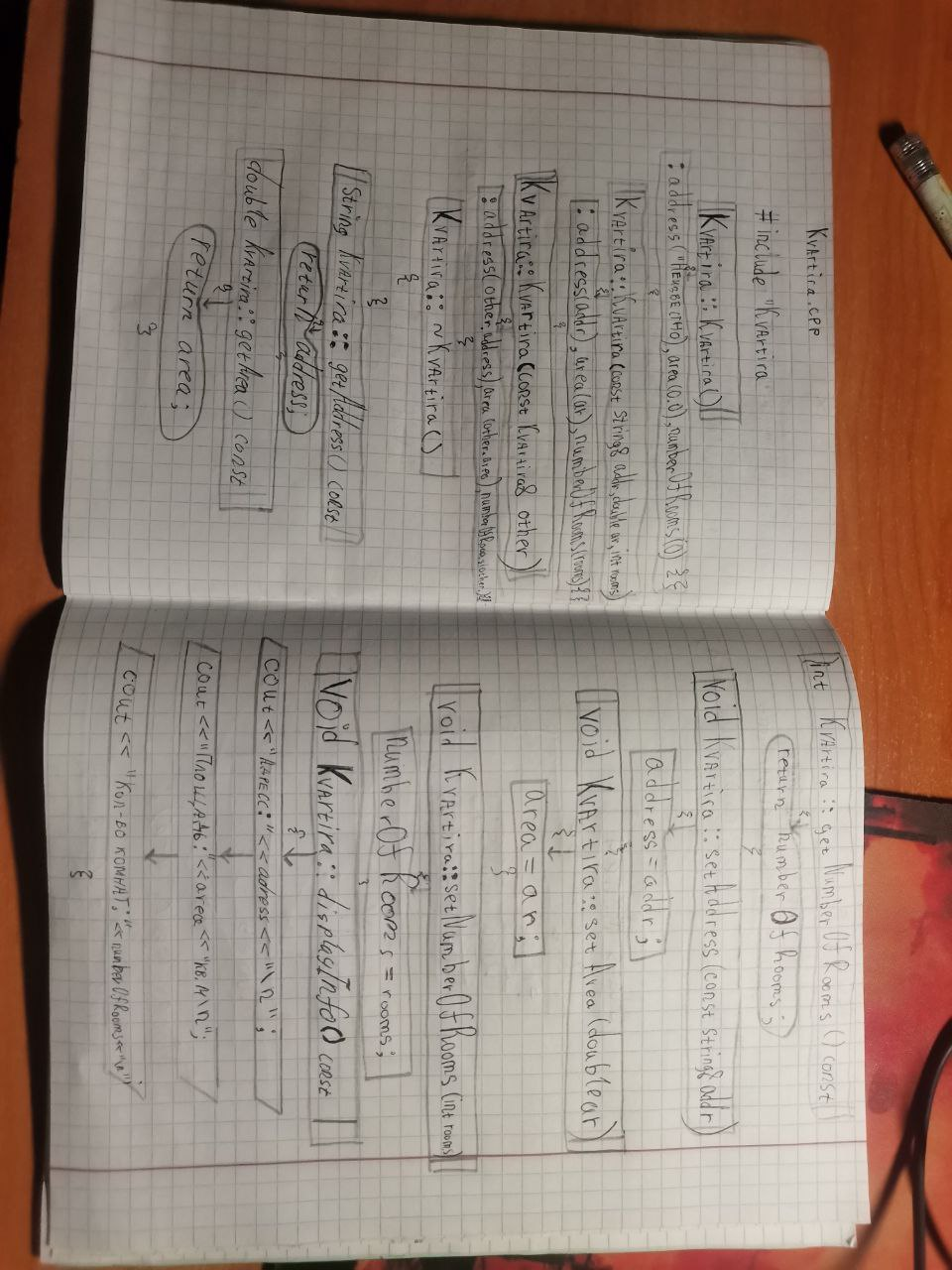
5. Написать демонстрационную программу, в которой продемонстрировать все три случая вызова конструктора-копирования, вызов конструктора с параметрами и конструктора без параметров.

Персональная:

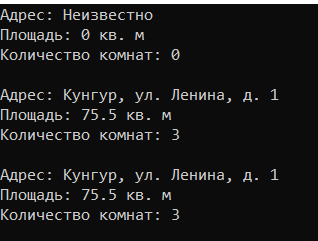


Блок-схема:





Результат работы:



Контрольные вопросы:

1. Для чего нужен конструктор?

2. Сколько типов конструкторов существует в С++?

3. Для чего используется деструктор? В каких случаях деструктор описывается явно?

4. Для чего используется конструктор без параметров? Конструктор с параметрами? Конструктор копирования?

5. В каких случаях вызывается конструктор копирования?

6. Перечислить свойства конструкторов.

7. Перечислить свойства деструкторов.

8. К каким атрибутам имеют доступ методы класса?

9. Что представляет собой указатель this?

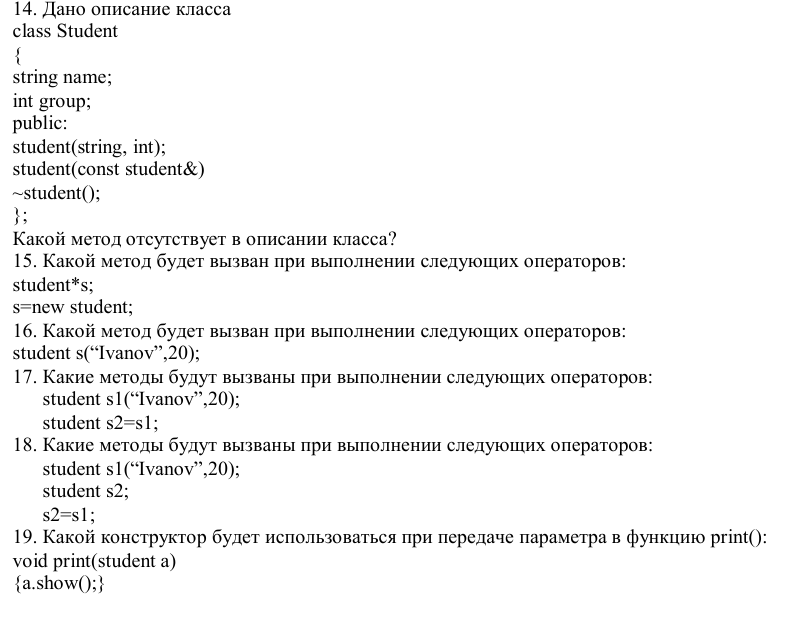
10. Какая разница между методами определенными внутри класса и вне

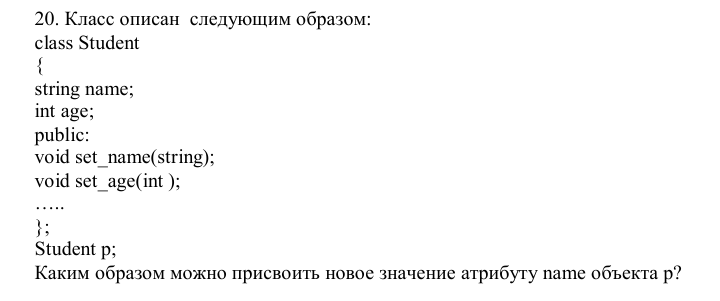
класса?

11. Какое значение возвращает конструктор?

12. Какие методы создаются по умолчанию?

13. Какое значение возвращает деструктор?





Ответы:

1. **Для чего нужен конструктор?**  
   Конструктор нужен для инициализации объекта, чтобы избежать ошибок, связанных с использованием неинициализированных переменных.
2. **Сколько типов конструкторов существует в С++?**  
   Существует три типа конструкторов: конструктор с параметрами, конструктор без параметров и конструктор копирования.
3. **Для чего используется деструктор? В каких случаях деструктор описывается явно?**  
   Деструктор используется для освобождения ресурсов, выделенных конструктором объекту. Деструктор описывается явно, если объект содержит указатели на динамически выделенную память, чтобы избежать утечек памяти.
4. **Для чего используется конструктор без параметров? Конструктор с параметрами? Конструктор копирования?**
   * Конструктор без параметров используется для создания «пустого» объекта.
   * Конструктор с параметрами используется для инициализации объекта требуемыми значениями.
   * Конструктор копирования используется для создания объекта, аналогичного уже существующему.
5. **В каких случаях вызывается конструктор копирования?**  
   Конструктор копирования вызывается при описании нового объекта с инициализацией другим объектом, при передаче объекта в функцию по значению и при возврате объекта из функции.
6. **Перечислить свойства конструкторов.**
   * Конструктор не возвращает значение, даже типа void.
   * Класс может иметь несколько конструкторов с разными параметрами (перегрузка).
   * Конструктор, вызываемый без параметров, называется конструктором по умолчанию.
   * Параметры конструктора могут иметь любой тип, кроме этого же класса.
   * Конструкторы не наследуются.
   * Конструкторы нельзя описывать с модификаторами const, virtual и static.
7. **Перечислить свойства деструкторов.**
   * Деструктор не имеет аргументов и возвращаемого значения.
   * Деструктор не наследуется.
   * Деструктор не может быть объявлен как const или static, но может быть виртуальным.
   * Если деструктор не определен, компилятор автоматически создает пустой деструктор.
8. **К каким атрибутам имеют доступ методы класса?**  
   Методы класса имеют неограниченный доступ ко всем элементам класса, независимо от спецификаторов доступа и порядка объявления методов в классе.
9. **Что представляет собой указатель this?**  
   Указатель this представляет собой указатель на текущий объект, для которого вызывается метод.
10. **Какая разница между методами, определенными внутри класса и вне класса?**  
    Методы, определенные внутри класса, считаются встроенными (inline), в то время как методы, определенные вне класса, должны быть связаны с именем класса.
11. **Какое значение возвращает конструктор?**  
    Конструктор не возвращает значение.
12. **Какие методы создаются по умолчанию?**  
    По умолчанию создаются конструктор без параметров и конструктор копирования (если не указаны явно).
13. **Какое значение возвращает деструктор?**  
    Деструктор не возвращает значение.
14. **Какой метод отсутствует в описании класса?**

* Отсутствует конструктор без параметров.

1. **Какой метод будет вызван при выполнении следующих операторов:**

* student \*s;
* Указатель s создается, но объект не инициализируется, поэтому метод не вызван (если нет конструктора без параметров).

1. **Какой метод будет вызван при выполнении следующих операторов:**

* student s("Ivanov", 20);
* Будет вызван конструктор с параметрами student(string, int).

1. **Какие методы будут вызваны при выполнении следующих операторов:**

* student s1("Ivanov", 20);
* student s2 = s1;
* Вызовется конструктор с параметрами student(string, int) для s1 и конструктор копирования student(const student&) для s2.

1. **Какие методы будут вызваны при выполнении следующих операторов:**

* student s1("Ivanov", 20);
* student s2;
* s2 = s1;
* Вызовется конструктор с параметрами student(string, int) для s1, конструктор без параметров (по умолчанию) для s2 и конструктор копирования для s2 = s1.

1. **Какой конструктор будет использоваться при передаче параметра в функцию print()?**

* Будет использоваться конструктор копирования student(const student&), так как объект передается по значению.

1. Для присвоения нового значения атрибуту name объекта p можно использовать метод set\_name(string), определенный в классе Student. Пример присвоения:

p.set\_name("Новое имя");

Ссылка:

**https://github.com/LeonidZhir/-**