ПНИПУ

Пермский национальный исследовательский политехнический университет

Лабораторная работа ООП 2

Выполнил студент ИВТ-23-2б

Меновщиков Глеб Николаевич

Проверила:

Доцент кафедры ИТАС

Ольга Андреевна Полякова

2024

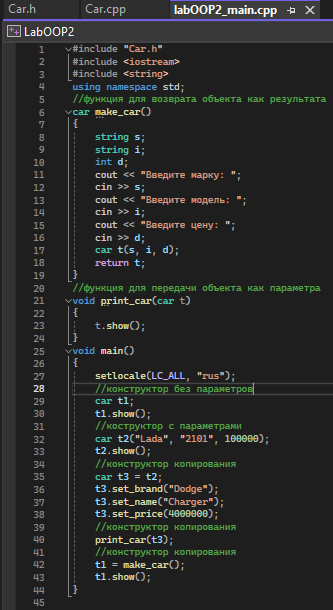
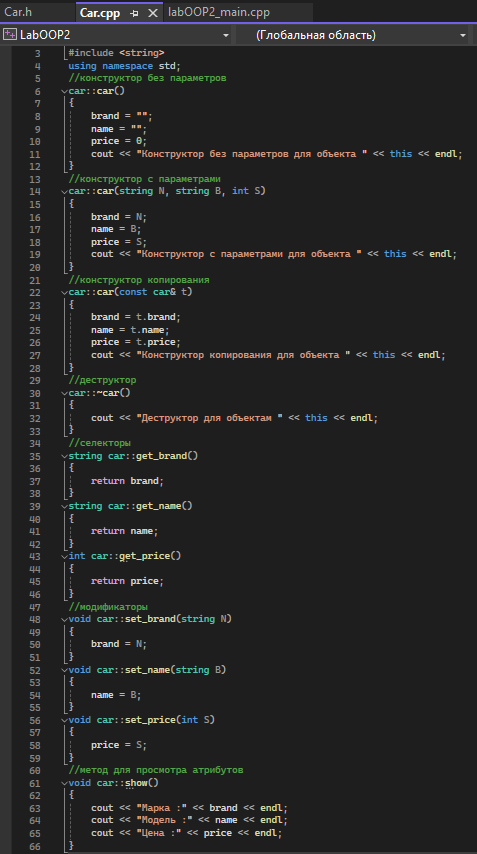
**11 Вариант**.

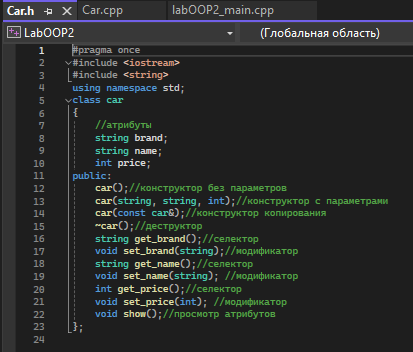
Пользовательский класс АВТОМОБИЛЬ

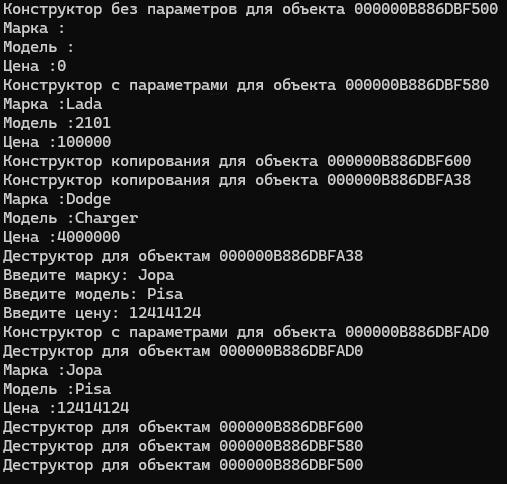
Марка – string

Модель – string

Стоимость– int





Вопросы:

1. Для чего нужен конструктор? это специальный метод класса, который автоматически вызывается при создании объекта класса.

2. Сколько типов конструкторов существует в С++? 4

3. Для чего используется деструктор? В каких случаях деструктор описывается явно? это специальный метод класса, который автоматически вызывается при уничтожении объекта. Деструктор описывается явно в случае, если класс имеет специфические ресурсы, которые нужно корректно освободить при уничтожении объекта.

4. Для чего используется конструктор без параметров? Конструктор с параметрами? Конструктор копирования?1 Когда требуется создать объект с начальными значениями атрибутов по умолчанию без необходимости предоставления пользовательских значений. 2 Когда требуется создать объект с определенными значениями атрибутов, переданными как параметры конструктора. 3 Когда требуется создать новый объект как точную копию существующего объекта.

5. В каких случаях вызывается конструктор копирования? Копирование объекта при инициализации Передача объекта по значению в функцию Возврат объекта из функции по значению Динамическое выделение памяти

6. Перечислить свойства конструкторов.

1. Конструктор и класс имеют одинаковое имя.
2. Конструктор можно перегружать(класс может иметь несколько конструкторов)
3. Конструктор не возвращает значение.
4. При отсутствии конструктора компилятор сам создает конструктор.
5. Конструктор должен быть **public**

7. Перечислить свойства деструкторов. Имеет одно имя с классом и объявляется с помощью ~

8. К каким атрибутам имеют доступ методы класса? Методы класса в C++ имеют доступ ко всем членам класса

9. Что представляет собой указатель this? содержит адрес объекта, который вызывает метод класса.

10. Какая разница между методами определенными внутри класса и вне класса?

Методы, определенные внутри класса (Inline-методы):

* Автоматически инлайн, компилятор пытается вставить код непосредственно в вызывающий код.
* Прямой доступ ко всем членам класса.
* Может увеличить время компиляции при определении в заголовочном файле.

Методы, определенные вне класса:

* Разделение интерфейса и реализации, улучшает читаемость и ускоряет компиляцию.
* Требует явного указания класса через оператор **::**.
* Нет прямого доступа к приватным и защищенным членам класса, доступ через публичный интерфейс.

11. Какое значение возвращает конструктор? не возвращает

12. Какие методы создаются по умолчанию? Конструкторы, деструктор, оператор присваивания.

13. Какое значение возвращает деструктор? не возвращает

14. Дано описание класса

class Student

{

string name; int group; public:

student(string, int); student(const student&)

~student();

};

Какой метод отсутствует в описании класса? оператора присваивания

15. Какой метод будет вызван при выполнении следующих операторов: student\*s;

s=new student; конструктор по умолчанию

16. Какой метод будет вызван при выполнении следующих операторов: student s(“Ivanov”,20); конструктор с параметрами класса **Student**

17. Какие методы будут вызваны при выполнении следующих операторов: student s1(“Ivanov”,20);

student s2=s1; конструктор с параметрами, конструктор с параметрами

18. Какие методы будут вызваны при выполнении следующих операторов: student s1(“Ivanov”,20);

student s2; s2=s1; конструктор с параметрами конструктор по умолчанию оператор присваивания

19. Какой конструктор будет использоваться при передаче параметра в функцию print(): void print(student a)

{a.show();}

конструктор копирования

20. Класс описан следующим образом:

class Student

{

string name; int age; public:

void set\_name(string); void set\_age(int );

…..

};

Student p;

Каким образом можно присвоить новое значение атрибуту name объекта р?

p.set\_name("Новое имя");