МИНИСТЕРСТВО НАУКИ и высшего образования

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

**Отчёт по лабораторной работе № 7**

**по учебной дисциплине «Системное программирование»**

**Тема: «Программирование виртуальных функций при наследовании классов на языке C++, абстрактные классы»**

Выполнил(а) студент(ка)

специальности 09.02.07

Информационные системы и

программирование

IV курса группы 42919/7

Побирчев Никита Андреевич

Преподаватель

Молькова Лолита Юрьевна

Санкт-Петербург,

2024

**Лабораторная работа №7**

**Цель работы:**

Изучить возможности виртуальных функций при наследовании классов на языке С++, абстрактные классы.

**Задание:**

1. Определить иерархию классов (в соответствии с вариантом).
2. Определить в классе статическую компоненту - указатель на начало связанного списка объектов и статическую функцию для просмотра списка.
3. Реализовать классы.
4. Написать демонстрационную программу, в которой создаются объекты различных классов и помещаются в список, после чего список просматривается.
5. Сделать соответствующие методы не виртуальными и посмотреть, что будет.
6. Реализовать вариант, когда объект добавляется в список при создании, т.е. в конструкторе.

**Индивидуальное задание:**

Республика, монархия, королевство, государство.

**Ход работы:**

Создал класс State (Рисунок 1).

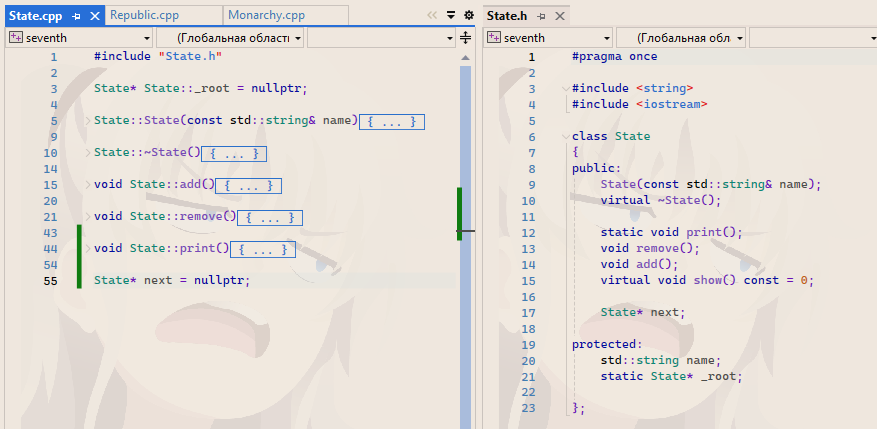


Рисунок 1 - Класс State

Реализация метода add (Рисунок 2).

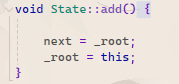


Рисунок 2 - Метод add.

Реализация метода remove (Рисунок 3).

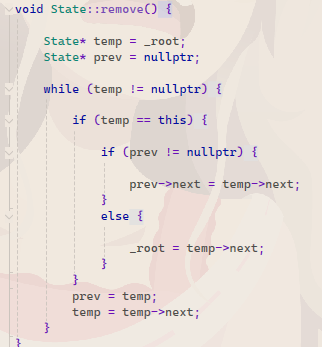


Рисунок 3 - Метод remove.

Реализация вывода списка (Рисунок 4).

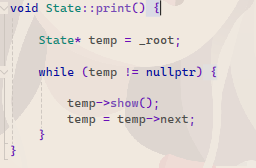


Рисунок 4 - Метод print.

Указатель на следующий элемент в списке (Рисунок 5).



Рисунок 5 - Указатель на следующий элемент в списке.

Указатель на корневой элемент списка (Рисунок 6).



Рисунок 6 - Указатель на корневой элемент списка.

Пример реализации производных классов (Рисунок 7).

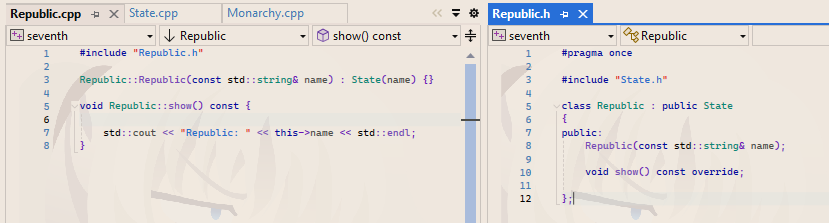


Рисунок 7 - Реализация класса Republic.

В функции main(), создал экземпляры 3-х дочерних классов и вызвал метод вывода информации о всех объектах (Рисунок 8).



Рисунок 8 - Функция main.

Результат работы программы (Рисунок 9).

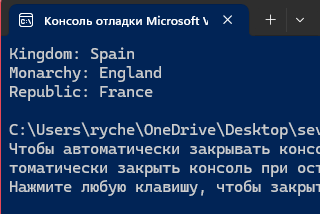


Рисунок 9 - Результат работы программы.

**Контрольные вопросы.**

1. **Какие возможности перед программистом открывают виртуальные функции?**

Виртуальные функции позволяют реализовать полиморфизм, что означает возможность обработки объектов разных типов через один интерфейс. Это позволяет выбирать метод в зависимости от типа объекта, на который указывает указатель или ссылка, а не от типа самого указателя.

1. **Истинно ли утверждение о том, что указатель на базовый класс может ссылаться на объекты порожденного класса?**

Истинно.

1. **Пусть указатель p ссылается на объекты базового класса и содержит адрес объекта порожденного класса. Пусть в обоих этих классах имеется невиртуальный метод ding(). Тогда выражение p->ding() вызовет метод ding() из ......... класса.**

Базового класса.

1. **Напишите описание для виртуальной функции dang(), возвращающей результат void и имеющей аргумент типа int.**

virtual void dang(const int&);

1. **Пусть указатель p ссылается на объекты базового класса и содержит адрес объекта порожденного класса. Пусть в обоих этих классах имеется виртуальный метод ding(). Тогда выражение p->ding() вызовет метод ding() из ......... класса.**

Дочернего класса.

1. **Напишите описание для чистой виртуальной функции aragorn(), не возвращающей значений и не имеющей аргументов.**

virtual void aragorn() = 0;

1. **Чистая виртуальная функция, это виртуальная функция, которая:** делает свой класс абстрактным.
2. **Напишите определение массива parr, содержащего 10 указателей на объекты класса dong.**

dong\* parr[10];

1. **Абстрактный класс используется тогда,** когда необходимо запретить создавать объекты класса.