**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра математического обеспечения и применения ЭВМ**

**ОТЧЕТ**

**по лабораторной работе №4**

**по дисциплине «Объектно-ориентированное программирование»**

**Тема: Умные указатели**

Студент гр. 7303 Батурин И.

Преподаватель Размочаева Н.В.

Санкт-Петербург

2019

**Цель работы.**

Реализация умного указателя разделяемого владения объектом, с функциями, аналогичными по функционалу std::shared\_ptr.

**Постановка задачи.**

Реализовать конструкторы, операторы, методы собственного класса умного указателя shared\_ptr для возможности его применения в коде, с поведением, аналогичным std::shared\_ptr. Реализованный класс должен также иметь поддержку полиморфизма, а именно: копирование указателей на полиморфные объекты и сравнение умных указателей как указателей на хранимые объекты.

**Выполнение работы.**

Класс shared\_ptr содержит два приватных поля: ptr и counter – указатели на хранимый объект и на счетчик количества умных указателей на тот же объект. Ниже приведен порядок реализации публичных функций класса.

1. Класс имеет два конструктора: принимающий стандартный указатель и конструктор копирования, а также деструктор.
2. Реализованы оператор присваивания (копирующий), оператор приведения типа bool() (true, если ненулевой указатель), операторы \*

и ->, возвращающие ссылку и указатель на хранимый объект соответственно.

1. Реализованы методы use\_count, возвращающий число указателей на хранимый объект; swap, меняющий содержимое заданного и текущего указателя; reset, заменяющий текущий указатель на объект заданным.
2. Для поддержки полиморфизма модифицированы конструктор копирования, оператор присваивания, а также реализован оператор ==, это сделано путем объявления дружественным класса shared\_ptr с

другим типом указателя, и добавлением этой сигнатуры к модифицированным и созданным функциям.

**Выводы.**

В ходе работы на языке C++ был реализован умный указатель разделяемого владения объектом shared\_ptr, с поддержкой полиморфного использования.