**Университет ИТМО**

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.04.01 Информатика и вычислительная техника

Дисциплина «Программирование на C++»

**Отчет**

По лабораторной работе №2

Выполнил:

*Казаченко Р. О.*

Преподаватель:

*Лаздин А. В.*

Санкт-Петербург, 2024 г.

**Вариант задания:**

Необходимо спроектировать класс, реализующий хранение данных, связанных с экземпляром класса в динамической памяти. Это может быть, например представление целых чисел, для которых помимо значения типа int хранится строковое представление числа (22 и «двадцать два»). Пример условен.

Для данного класса необходимо реализовать все необходимые конструкторы, включая конструкторы копирования и перемещения, деструктор. Все конструкторы и деструктор должны «сообщать» о своём вызове. std::cout << “Copy constructor” << std::endl; то же относится и к перегрузке операции присваивания (два варианта: без перемещения и с перемещением). Определение класса должно быть помещено в заголовочный файл (.hpp), реализация методов в отдельном файле .cpp.

Написать программу (ещё один модуль cpp) осуществляющую работу с экземплярами разработанного класса, которая должна демонстрировать:

• Создание статических и динамических экземпляра класса, стандартного вектора в стиле Си, написать функцию для обработки данного вектора;

• Передача экземпляров класса в функцию, и возврат экземпляра из функции; включая вариант с передачей и возвратом значений типа ссылки на класс;

• Продемонстрировать работу с разработанным классом, создавая вектора и списки экземпляров класса, используя стандартные классы vector и list длиной от 5 до 10.

Проанализировать процессы создания и удаления экземпляров класса для различных примеров. Необходимо получить вывод от всех конструкторов, деструктора и перегруженных операций присваивания.

**Исходный код:**

<https://github.com/R4MZEZ/SiPlusPlus>

**Примеры работы:**

*Тест создания статических и динамических экземпляра класса:*

|  |  |
| --- | --- |
|  |  |
|  |  |

*Тест создания вектора экземпляров класса:*

|  |  |
| --- | --- |
|  |  |
|  |  |

*Тест создания списка экземпляров класса:*

|  |  |
| --- | --- |
| *Изображение выглядит как текст, снимок экрана, меню, Шрифт  Автоматически созданное описание* | *Изображение выглядит как текст, снимок экрана, меню, Шрифт  Автоматически созданное описание* |

*Тест передачи и возврата экземпляра класса в функцию по значению:*

*Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание*

*Тест передачи и возврата экземпляра класса в функцию по ссылке:*

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

**Выводы:**

Изучил механизм вызова конструкторов и деструкторов, а также механизм их вызова

в различных сценариях обработки объектов.