МИНОБРНАУКИ РОССИИ

ФГБОУ ВО Череповецкий государственный университет

Институт информационных технологий

Кафедра: МПО ЭВМ

Дисциплина: Объектно-ориентированное Программирования

ЛАБОРАТОРНАЯ РАБОТА № 1

# Тема: Разработка абстрактных типов данных

Выполнил: Минеев Иван Александрович

студент гр.: 1ПИб-03-22ОП

Проверил: доцент, к.т.н.

Табунов Павле Александрович

Череповец, 2023 г.

**ЦЕЛЬ РАБОТЫ**

Изучить способы определения классов, правила доступа к элементам; приобрести практические навыки работы с объектами класса; изучить принципы и механизмы создания абстрактных типов данных.

Изучить синтаксис и семантику определения и вызова конструкторов и деструкторов. Приобрести практические навыки создания и уничтожения объектов. Изучить особенности применения различных видов конструкторов.

**ЗАДАНИЯ**

Разработайте алгоритм и программу, реализующую абстрактный тип данных (АТД) – класс, согласно варианту задания (см. прил. 1):

1. Предусмотрите закрытую реализацию и открытый интерфейс.
2. Интерфейс должен содержать функции вывода содержимого и функции доступа к некоторым полям.
3. Элементы данных должны быть представлены из 5 полей, один из которых должен быть статическим, второй иметь тип указатель, а третий тип перечисление, остальные на свое усмотрение.
4. Примените основные виды конструкторов и определитесь с деструктором класса.
5. Проверьте работоспособность АТД на тестовом наборе данных.

**ХОД РАБОТЫ**

**КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Дайте определение понятия «класс». Сформулируйте правила доступа к его элементам.

Класс – абстрактный тип данных, позволяющий создать свой собственный тип данных и способы его обработки. По умолчанию элементы структуры являются открытыми, элементы класса – закрытыми.

2. С какой целью в классе объединены компонентные данные и компонентные функции?

Чтобы иметь доступ ко всем элементам класса.

3. Каким образом осуществляется доступ к открытым и закрытым элементам?

К открытым элементам доступ осуществляется через . или ->, к закрытым элементам через открытые функции.

4. Опишите назначение дружественных функций, назовите их разновидности.

Дружественными могут быть внешними и компонентным, нужны для получения доступа к закрытым элементам класса на ряду с компонентными.

5. Что понимается под указателем this?

Ключевое слово this неявно объявляет указатель на себя. Может иметь место только в нестатических компонентных функциях.

6. Каковы особенности использования статических компонентных данных?

Статические элементы данных хранятся в одном месте, являются общими для всех объектов класса и существуют независимо от них. Статическая компонентная функция доступа к компонентным данным не имеет, за исключением статических.

7. В чем заключается синтаксис и семантика компонентных функций static и const?

Ключевые слова static и const ставятся перед объявлением данных или функций. static для данных означает, что объект общий, а для функций означает, что функция имеет доступ лишь к статическим данным. При сonst данные не могут быть изменены, а в функции не могут быть изменены значения переменных.

8. Каким образом могут изменяться компонентные данные объектов, объявленных константами?

Отказаться от константного значения можно с помощью const\_cast или индивидуально для отдельных компонентных данных с использованием ключевого слова mutable.

9. Каковы особенности создания вложенных классов?

Вложение классов позволяет вкладывать определения функций - компонентная функция должна быть определена внутри своего локального класса, и на данную функцию нельзя ссылаться вне области видимости ее локального класса.