Министерство образования и науки Российской Федерации

Федеральное государственное образовательное автономное учреждение высшего образования

"Пермский национальный исследовательский политехнический университет"

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №8

Дисциплина: Основы алгоритмизации и программирования

Тема: Программа, управляемая событиями

Вариант 13

|  |  |
| --- | --- |
| Выполнил работу | |
| Студент группы РИС-22-1б | |
| Кирпичников Илья | |
|  | |
| Проверил работу | |
| Доцент кафедры ИТАС | |
| Полякова О.А. | |
|  | |

Пермь – 2023

**Постановка задачи**

1. Определить иерархию пользовательских классов (см. лабораторную работу №5). Во главе иерархии должен стоять абстрактный класс с чисто виртуальными методами для ввода и вывода информации об атрибутах объектов.
2. Реализовать конструкторы, деструктор, операцию присваивания, селекторы и модификаторы.
3. Определить класс-группу на основе структуры, указанной в варианте.
4. Для группы реализовать конструкторы, деструктор, методы для добавления и удаления элементов в группу, метод для просмотра группы, перегрузить операцию для получения информации о размере группы.
5. Определить класс Диалог – наследника группы, в котором реализовать методы для обработки событий.
6. Добавить методы для обработки событий группой и объектами пользовательских классов.
7. Написать тестирующую программу.
8. Нарисовать диаграмму классов и диаграмму объектов.

Вариант 15:

Базовый класс:

* ПЕЧАТНОЕ\_ИЗДАНИЕ(PRINT)
* Название – string
* Автор – string

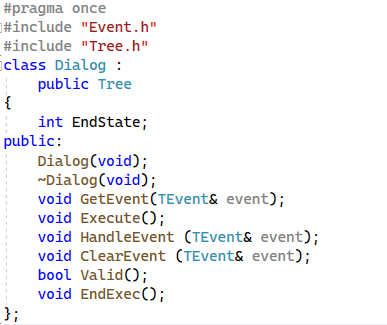
Производный класс

* ЖУРНАЛ (MAGAZIN)
* Количество страниц - int
* Группа – Дерево (Tree)

Команды:

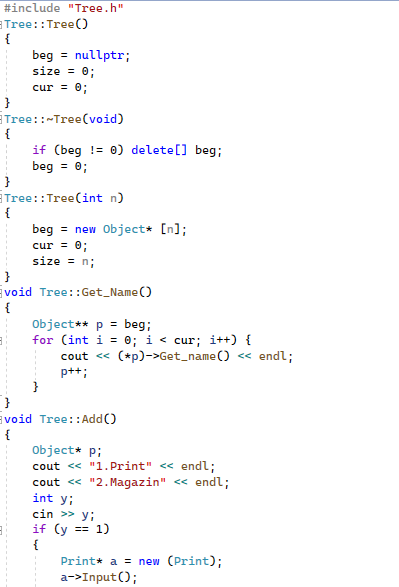
* Создать группу (формат команды: m количество элементов группы).
* Добавить элемент в группу (формат команды: +)
* Удалить элемент из группы (формат команды -)
* Вывести информацию об элементах группы (формат команды: s)
* Вывести информацию о названиях всех элементов группы (формат команды : z) Конец работы (формат команды: q)

# Описание класса-контейнера

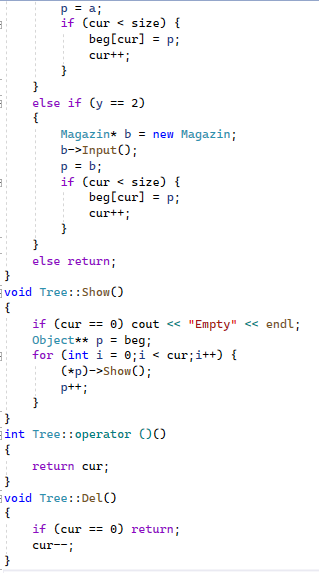
****

*Приложение 1 – Описание класса-контейнера*

# Определение компонентных функций

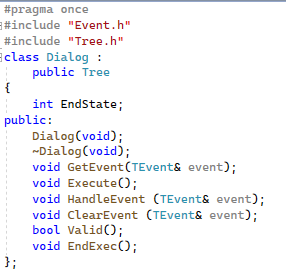
****

*Приложение 2 – Определение компонентных функций*

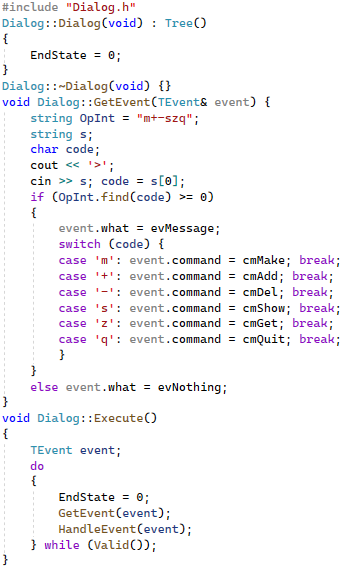
****

*Приложение 3 – Определение компонентных функций*

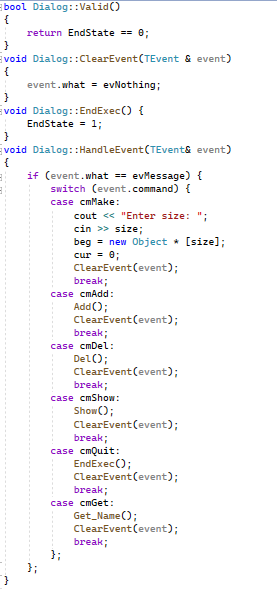
**Описание класса-диалога и его компонентных функций**

****

*Приложение 4 – Описание класса-диалога и его компонентных функций*

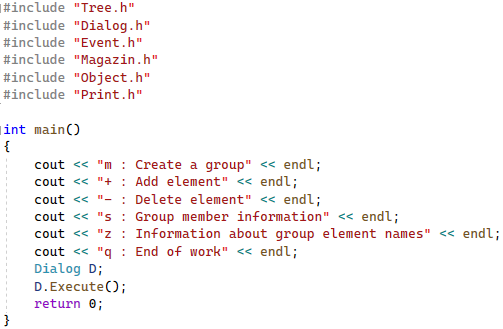
****

*Приложение 5 – Описание класса-диалога и его компонентных функций*

****

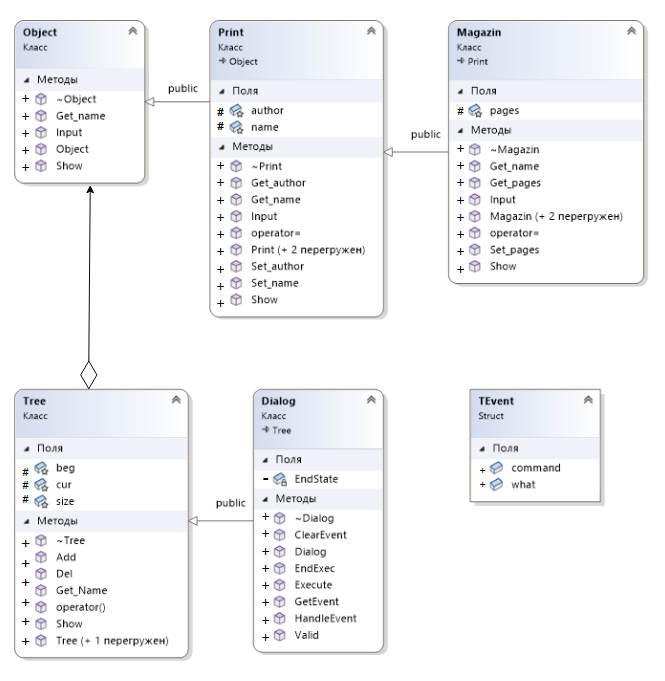
*Приложение 6 – Описание класса-диалога и его компонентных функций*

**Функция main()**

****

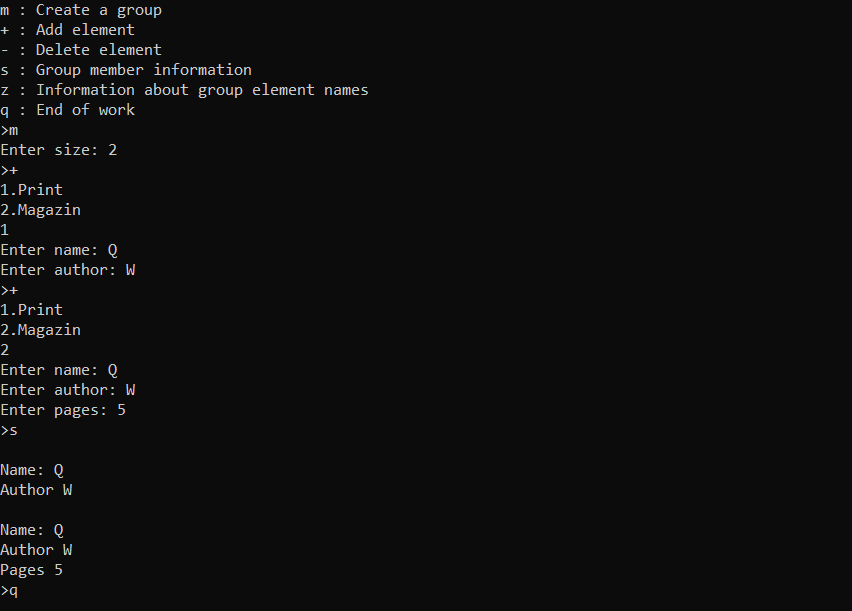
*Приложение 7 – Главная функция*

**UML-Диаграмма**



*Приложение 8 – UML-Диаграмма*

**Работа программы**



*Приложение 9 – Работа программы*

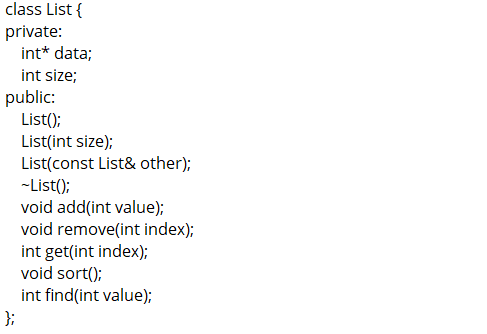
**Ответы на контрольные вопросы**

1.Что такое класс-группа? Привести примеры таких классов.

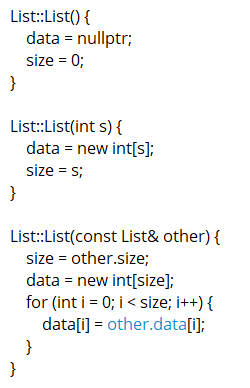
Класс-группа - это группа классов, имеющих общие свойства и методы. Примеры таких классов: коллекции (List, Set, Map), геометрические фигуры (Circle, Rectangle, Triangle), транспортные средства (Car, Bus, Train).

2.Привести пример описания класса-группы Список (List).

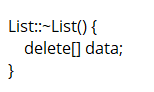
Описание класса-группы Список может выглядеть так: класс List содержит методы для добавления, удаления и просмотра элементов списка, а также методы для сортировки и поиска элементов.



3.Привести пример конструктора (с параметром, без параметров, копирования) для класса-группы Список.

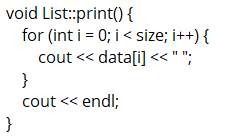


4.Привести пример деструктора для класса-группы Список.



Пример деструктора для класса-группы Список: ~List() - освобождает память, занятую элементами списка.

5.Привести пример метода для просмотра элементов для класса-группы Список.



Пример метода для просмотра элементов для класса-группы Список: void print() - выводит все элементы списка на экран.

6.Какой вид иерархии дает группа?

Группа дает иерархию, которая позволяет организовать классы в древовидную структуру, где каждый класс наследует свойства и методы от более общего класса.

7.Почему во главе иерархии классов, содержащихся в группе объектов должен находиться абстрактный класс?

Абстрактный класс должен находиться во главе иерархии, потому что он определяет общие свойства и методы для всех классов в группе, но не имеет конкретной реализации. Он служит основой для создания конкретных классов-наследников.

8.Что такое событие? Для чего используются события?

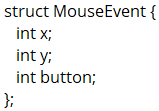
Событие - это сигнал от объекта о том, что произошло какое-то событие (например, нажатие кнопки мыши). События используются для управления программой и обработки пользовательского ввода.

9.Какие характеристики должно иметь событие-сообщение?

Событие-сообщение должно содержать информацию о типе события, объекте, который его вызвал, и дополнительные данные (например, координаты мыши).

10.Привести пример структуры, описывающей событие.

Пример структуры, описывающей событие:



Эта структура содержит информацию о координатах мыши и кнопке, которая была нажата.

11.Задана структура события. Какие значения, и в каких случаях присваиваются полю what?

Поле what принимает значения, которые указывают на тип события, например: 1 - клик мыши, 2 - нажатие клавиши, 3 - сообщение.

12.Задана структура события. Какие значения, и в каких случаях присваиваются полю command?

Полю command присваиваются значения кодов команд, например: 1 - открыть файл, 2 - сохранить файл, 3 - закрыть программу.

13.Задана структура события. Для чего используются поля a и message?

Поля message и a используются для передачи параметров команды, например: при команде открытия файла поле message может содержать путь к файлу, а поле a - режим открытия (чтение или запись).

14.Какие методы необходимы для организации обработки сообщений?

Методы для организации обработки сообщений могут включать методы для получения сообщений из очереди, методы для обработки конкретных типов сообщений и методы для отправки ответных сообщений.

15.Какой вид имеет главный цикл обработки событий-сообщений?

Главный цикл обработки событий-сообщений имеет вид бесконечного цикла, который получает сообщения из очереди и вызывает соответствующие методы для их обработки.

16.Какую функцию выполняет метод ClearEvent()? Каким образом?

Метод ClearEvent() очищает содержимое объекта события, устанавливая поля в значения по умолчанию.

17.Какую функцию выполняет метод HandleEvent ()? Каким образом?

Метод HandleEvent() выполняет обработку конкретного события, вызывая соответствующие методы и отправляя ответные сообщения.

18.Какую функцию выполняет метод GetEvent ()?

Метод GetEvent() используется для получения следующего события из очереди.

19.Для чего используется поле EndState? Какой класс (объект) содержит это поле?

Поле EndState используется для указания состояния завершения программы, например: 0 - успешное завершение, 1 - ошибка. Это поле может содержаться в объекте класса, который управляет выполнением программы.

20.Для чего используется функция Valid()?

Функция Valid() используется для проверки корректности данных или состояния объекта. Она может возвращать булево значение true, если объект корректен, и false, если есть ошибки или некорректные данные.