Министерство образования и науки Российской Федерации

Федеральное государственное образовательное автономное учреждение высшего образования

"Пермский национальный исследовательский политехнический университет"

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

Дисциплина: Основы алгоритмизации и программирования

Тема: Классы и объекты. Инкапсуляция.

Вариант: 13

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

Пермь – 2023

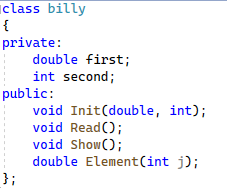
**Постановка задачи**

1. Реализовать определение нового класса.
2. Для демонстрации работы с объектами написать главную функцию. Продемонстрировать разные способы создания объектов и массивов объектов.
3. Структура-пара – структура с двумя полями, которые обычно имеют имена first и second.
4. Требуется реализовать тип данных с помощью такой структуры. Во всех заданиях должны присутствовать :
5. метод инициализации Init (метод должен контролировать значения аргументов на корректность);
6. ввод с клавиатуры Read;
7. вывод на экран Show.
8. Реализовать внешнюю функцию make\_тип(), где тип – тип реализуемой структуры.
9. Функция должна получать значения для полей структуры как параметры функции и возвращать структуру как результат.
10. При передаче ошибочных параметров следует выводить сообщение и заканчивать работу.

Вариант 15:

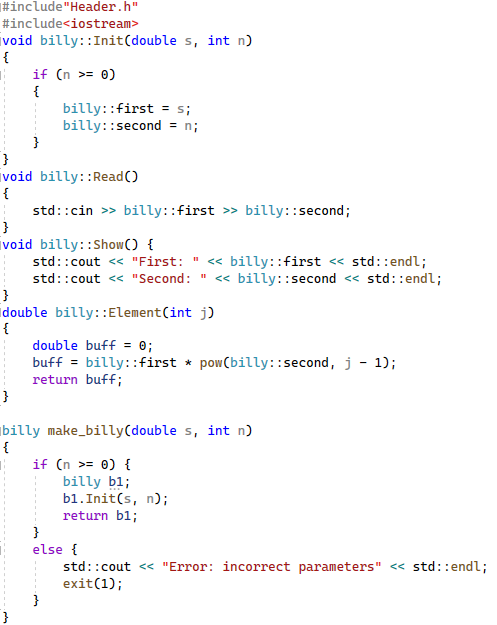
* Элемент арифметической прогрессии aj вычисляется по формуле: aj=a0·rj.
* Поле first – дробное число, первый элемент прогрессии a0, поле second – положительное целое число, постоянное отношение r.
* Реализовать метод element(int j) – вычисление j-го элемента прогрессии.

**Описание классов**

****

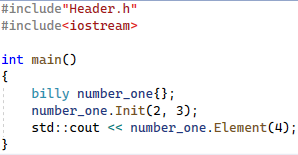
*Приложение 1 – Описание классов*

**Определение компонентных функций**

****

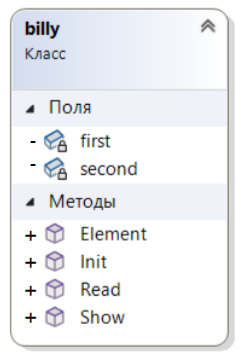
*Приложение 2 – Определение компонентных функций*

**Функция main()**

****

*Приложение 3 – Главная функция*

**UML-диаграмма**



*Приложение 4 – UML-диаграмма*

**Работа программы**



*Приложение 5 – Работа программы*

**Ответы на контрольные вопросы**

1. Класс – это пользовательский тип данных. Он определяет переменные (поля) и методы (функции), которые могут быть использованы объектами этого класса.

2. Объект (экземпляр) класса - это конкретный экземпляр класса, созданный на основе его шаблона. Он содержит свои собственные значения переменных (полей) и может вызывать методы (функции) определенные в классе.

3. Поля класса - это переменные, определенные внутри класса, которые хранят состояние объектов этого класса.

4. Функции класса - это методы, определенные внутри класса, которые могут быть вызваны объектами этого класса для выполнения определенных операций.

5. Спецификаторы доступа определяют уровень доступности к полям и методам класса из других частей программы.

6. Спецификатор public используется для определения методов и полей, которые могут быть доступны из любой части программы.

7. Спецификатор private используется для определения методов и полей, которые доступны только внутри самого класса.

8. Если описание класса начинается со спецификатора class, то по умолчанию будет использоваться спецификатор private.

9. Если описание класса начинается со спецификатора struct, то по умолчанию будет использоваться спецификатор public.

10. При описании интерфейса класса должен использоваться спецификатор public, так как интерфейс определяет открытые методы и свойства, доступные для использования в других частях программы.

11. Значения атрибутов экземпляра класса могут быть изменены путем вызова методов класса, которые изменяют значения переменных (полей).

12. Значения атрибутов экземпляра класса можно получить путем чтения значений переменных (полей) или вызова методов класса, которые возвращают значения.

13. Можно обратиться к полю name объекта s через оператора доступа к члену класса (.) следующим образом: s->name.

14. Можно обратиться к полю name объекта s через оператора доступа к члену класса (.) следующим образом: s.name.

15. Нельзя обратиться к полю name объекта s напрямую, так как оно не было инициализировано. Необходимо использовать методы класса для установки и получения значения поля name. Например, для установки значения можно использовать метод setName(), а для получения значения - метод getName():

s->setName("Иванов Иван");

string studentName = s->getName();

16. Нельзя обратиться к полям класса, так как оно не было находится в области private.

17. Можно обратиться к полю name объекта s используя оператор доступа к членам класса - точку (.):

s->name