Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра информационных технологий и автоматизированных систем

**ОТЧЕТ**

**Тема:** Лабораторная работа №1 по ООП

Семестр: 2

Выполнил студент ИВТ-23-2б:

Хомутов Сергей Алексеевич

(дата, подпись)

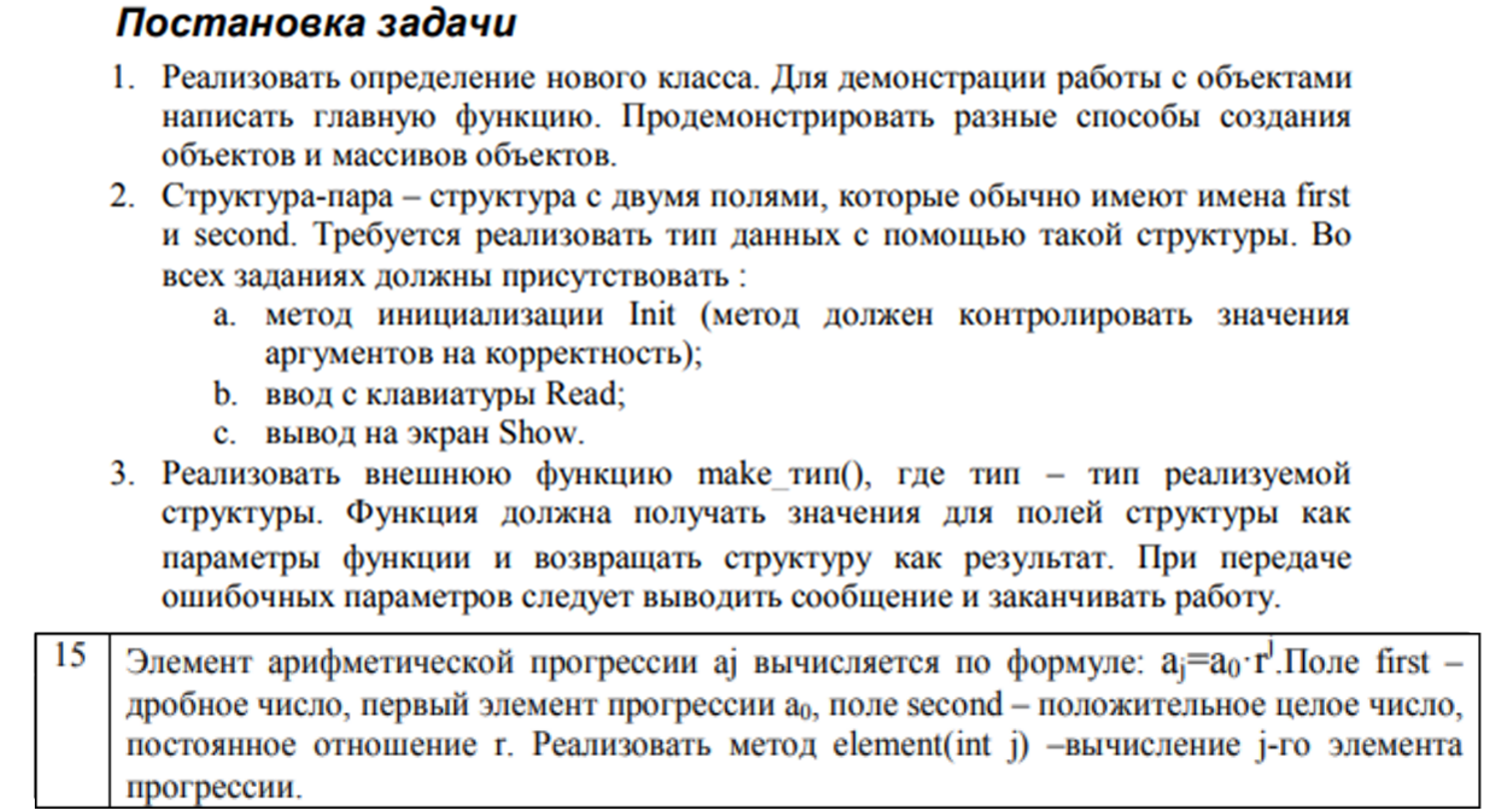
Проверила:

Полякова Ольга Андреевна

(дата, подпись)

Пермь 2024

**Задание**



**Анализ задачи**

1. Метод Init – инициализация переменных
2. Метод Show – вывод значений переменных
3. Метод Read – ввод значений переменных с клавиатуры
4. Метод Power – вычисление j-го элемента прогрессии

**UML-Диаграмма**

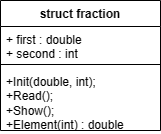
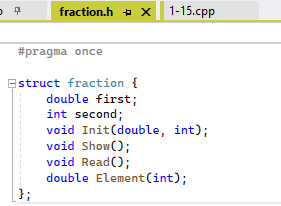
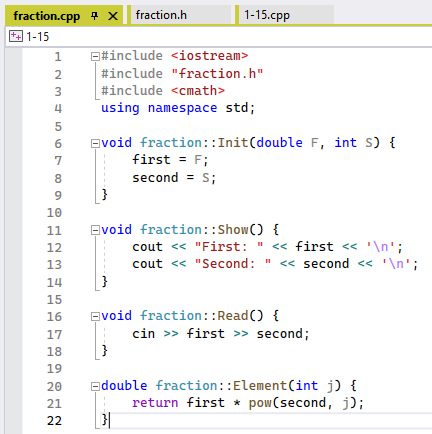


Рисунок 1 – UML-диаграмма структуры «fraction»

**Программное решение**







#include "fraction.h"

#include <iostream>

using namespace std;

fraction makeFraction(double, int);

int main() {

system("chcp 1251 > NULL");

//

fraction A;

A.Init(3.0, 2);

A.Show();

cout << A.Element(5) << '\n';

/\*другие способы:

fraction B;

B.Read();

B.Show();

int j1;

cout << "Введите номер какого элемента вы хотите получить " << '\n';

cout << "j1: ";

cin >> j1;

cout << B.Element(j1) << '\n';

fraction\* X = new fraction;

X->Init(2.0, 5);

X->Show();

fraction mas[3];

for (int i = 0; i < 3; i++) {

mas[i].Read();

}

for (int i = 0; i < 3; i++) {

mas[i].Show();

}

for (int i = 0; i < 3; i++) {

cout << " Степень? " << '\n';

cout << "j:";

cin >> j1;

mas[i].Element(j1);

}

fraction\* p\_mas = new fraction[3];

for (int i = 0; i < 3; i++) {

p\_mas[i].Read();

}

for (int i = 0; i < 3; i++) {

p\_mas[i].Show();

}

for (int i = 0; i < 3; i++) {

cout << " Степень? " << '\n';

cout << "j:";

cin >> j1;

cout << mas[i].Element(j1) << '\n';

}

double y; int z;

cout << "first:";

cin >> y;

cout << "second:";

cin >> z;

fraction F = makeFraction(y, z);

F.Show();

cout << F.Element(j1) << '\n';\*/

return 0;

}

fraction makeFraction(double f, int s) {

fraction t;

t.Init(f, s);

return t;

}



**Контрольные вопросы**

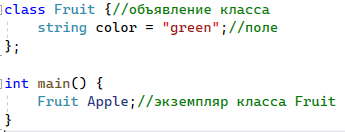
1. Что такое класс?

Класс - абстрактный пользовательский тип данных.

2. Что такое объект (экземпляр) класса?

Объектом называется отдельный элемент класса, который характеризуется данными его полей. Второе название объекта класса - экземпляр класса

Пример:



Фрукт – это класс.

Яблоко – это объект класса «Фрукт».

3. Как называются поля класса?

Атрибуты.

4. Как называются функции класса?

Методы.

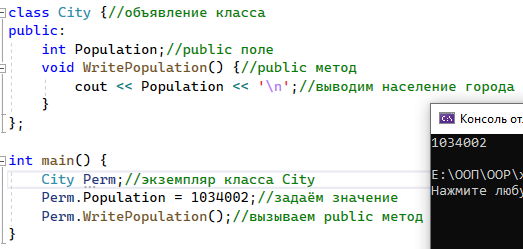
5. Для чего используются спецификаторы доступа?

Управление видимостью элементов класса.

6. Для чего используется спецификатор public?

Для общедоступных элементов класса

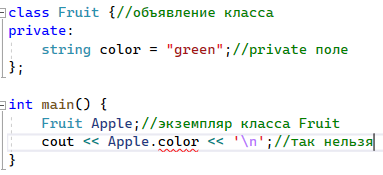
Пример:



7. Для чего используется спецификатор private?

Для закрытых элементов класса.

Пример:



8. Если описание класса начинается со спецификатора class, то

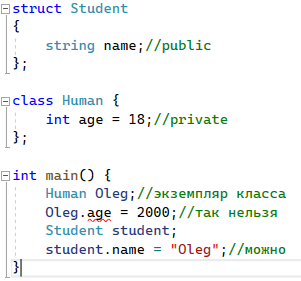
какой спецификатор доступа будет использоваться по умолчанию?

private

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

public

8-9 Пример:



10. Какой спецификатор доступа должен использоваться при описании интерфейса класса? Почему?

Public.

11. Каким образом можно изменить значения атрибутов экземпляра класса?

Через сеттер или напрямую.

12. Каким образом можно получить значения атрибутов экземпляра класса?

Через геттер или напрямую.

13. s->name;

14. s.name;

15. С помощью геттера.

16. С помощью геттера.

17. s->name.