



**«Московский государственный технический университет
имени Н.Э. Баумана»
(национальный исследовательский университет)
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ
КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

Отчет

по лабораторной работе № 8

Название лабораторной работы: Наследование.

Дисциплина: Алгоритмизация и программирование

Студент гр. ИУ6-14Б

 **02.12.2023**

(Подпись, дата)

А.С. Воеводин

(И.О. Фамилия)

Преподаватель

 **02.12.2023**

(Подпись, дата)

О.А. Веселовская

(И.О. Фамилия)

Москва, 2023

Цель работы

Решить поставленную задачу, используя механизмы наследования.

Задание

Разработать и реализовать иерархию классов для описанных объектов предметной области, используя механизмы наследования. Составить программный код и написать тестирующую программу.

Все поля классов должны быть частными (private) или защищенными (protected). Методы не должны содержать операций ввода/вывода, за исключением процедуры, единственной задачей которой является вывод информации об объекте на экран.

Объект – прямоугольный параллелепипед. Параметры: длина, ширина, высота. Методы: конструктор, процедура вывода параметров параллелепипеда на экран и функция определения объема параллелепипеда.

Объект – кирпич. Параметры: длина, ширина, высота, масса. Методы: конструктор, процедура вывода параметров кирпича на экран и функция определения плотности кирпича.

Ход работы:

- Написание программы.
- Тестирование программы при различных тестовых данных.
- Изображение схемы алгоритмов программы и диаграммы классов.
- Вывод.

Для начала напомним код программы, а именно: Заголовочные файлы Parallel.hpp и Brick.hpp, файлы реализации Parallel.cpp и Brick.cpp:

```

#ifndef LABA_8__PARALLEL_HPP_
#define LABA_8__PARALLEL_HPP_

class Parallel {
public:
    Parallel();
    Parallel(double len, double wid = 0.0, double heig = 0.0);
    void ShowInfo();
    double CalcVolume();

protected:
    double length;
    double width;
    double height;
};

#endif //LABA_8__PARALLEL_HPP_

```

Рисунок 1 – Заголовочный файл Parallel.hpp

```

#include "Parallel.hpp"
#include <iostream>

Parallel::Parallel() : length(0.0), width(0.0), height(0.0) {}
Parallel::Parallel(double len, double wid, double heig) : length(std::abs(x: len)), width(std::abs(x: wid)), height(std::abs(x: heig)) {}
void Parallel::ShowInfo() {
    std::cout << "Length: " << length << " Width: |" << width << " Height: " << height;
}
double Parallel::CalcVolume() {
    return length * width * height;
}

```

Рисунок 2 – Файл реализации Parallel.cpp

```

#ifndef LABA_8__BRICK_HPP_
#define LABA_8__BRICK_HPP_
#include "Parallel.hpp"

class Brick : public Parallel {
public:
    Brick();
    Brick(double len, double wid = 0.0, double heig = 0.0, double mass = 0.0);
    void ShowInfo();
    double CalcDensity();

private:
    double Mass;
};

#endif //LABA_8__BRICK_HPP_

```

Рисунок 3 – Заголовочный файл Brick.hpp

```

#include "Brick.hpp"
#include <iostream>

Brick::Brick() : Parallel(), Mass(0.0) {}
Brick::Brick(double len, double wid, double heig, double mass) : Parallel(len, wid, heig), Mass(std::abs(x: mass)) {}
void Brick::ShowInfo() {
    Parallel::ShowInfo();
    std::cout << " Mass: " << Mass;
}
double Brick::CalcDensity() {
    return Mass / CalcVolume();
}

```

Рисунок 4 – Файл реализации Brick.cpp

Далее создадим объекты классов Parallel и Brick:

```

#include <iostream>
#include "Brick.hpp"

int main() {
    Brick brick( len: 1, wid: 2, heig: 3, mass: 5);
    brick.ShowInfo();
    std::cout << std::endl;
    std::cout << brick.CalcVolume() << std::endl;
    std::cout << brick.CalcDensity() << std::endl << std::endl;

    Parallel par( len: 5, wid: 6, heig: 7);
    par.ShowInfo();
    std::cout << std::endl;
    std::cout << par.CalcVolume() << std::endl;

    return 0;
}

```

Рисунок 5 – Основной файл main.cpp

```

Length: 1 Width: 2 Height: 3 Mass: 5
6
0.833333

Length: 5 Width: 6 Height: 7
210

```

Рисунок 6 – Вывод программы

Как видно из рисунков, программа корректно выводит поля объектов, считает плотность и объём. Теперь изобразим программу в виде схемы и диаграммы классов:

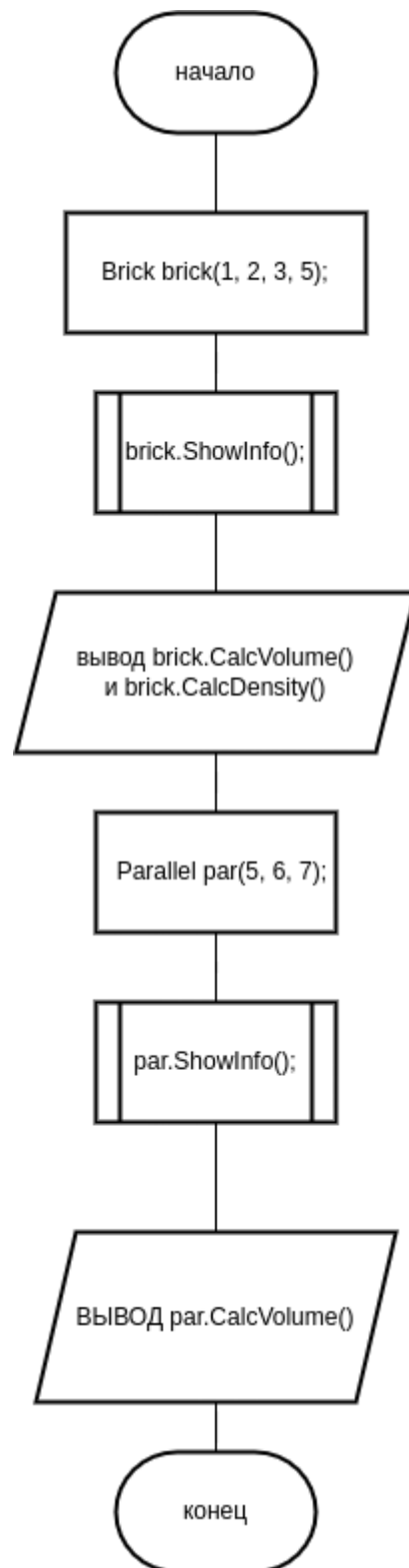


Рисунок 7 – Схема алгоритма основной программы

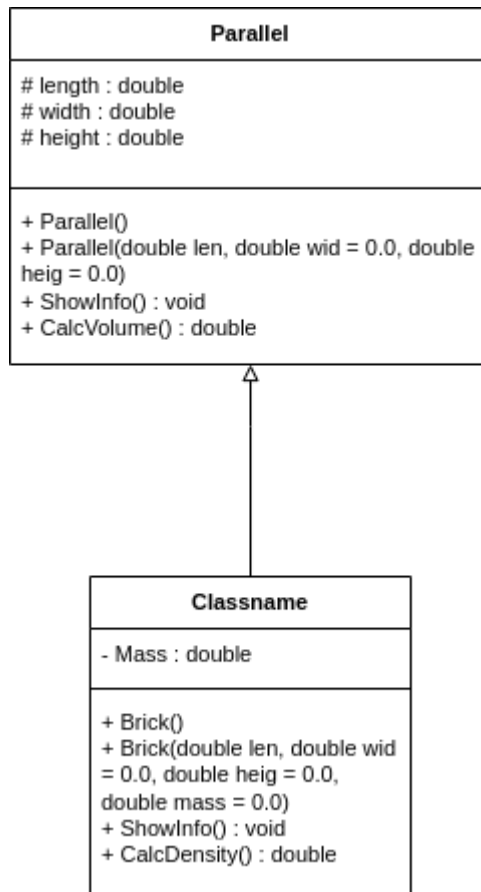


Рисунок 8 – Диаграммы классов, используемых в программе

Вывод

В ходе лабораторной работы были получены навыки работы с наследованием классов, инкапсуляцией их полей и переопределение конструкторов, методов.