

Московский Авиационный Институт
(Национальный Исследовательский Университет)

Кафедра: 806 "Вычислительная математика и программирование"
Факультет: "Информационные технологии и прикладная математика"
Дисциплина: "Объектно-ориентированное программирование"

Группа:
Студент: Пашкевич Андрей Романович
Преподаватель: Поповкин Александр Викторович

Москва, 2017

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1

Вариант №17

ЦЕЛЬ РАБОТЫ

Целью лабораторной работы является:

- Программирование классов на языке C++
- Управление памятью в языке C++
- Изучение базовых понятий ООП.
- Знакомство с классами в C++.
- Знакомство с перегрузкой операторов.
- Знакомство с дружественными функциями.
- Знакомство с операциями ввода-вывода из стандартных библиотек.

Задание:

Спроектировать и запрограммировать на языке C++ классы фигур (треугольник, квадрат, прямоугольник).

Классы должны иметь:

- Общий родительский класс Figure.
- Общий виртуальный метод Print, печатающий параметры фигуры и ее тип в стандартный поток вывода cout.
- Общий виртуальный метод расчета площади фигуры – Square.
- Конструктор, считывающий значения основных параметров фигуры из стандартного потока cin.

Классы должны быть расположены в отдельных файлах: отдельно заголовки (.h), отдельно описание методов (.cpp).

Программа должна позволять вводить фигуру каждого типа с клавиатуры, выводить параметры фигур и их площадь на экран.

Описание структуры классов и алгоритма работы программы:

Triangle.cpp	
Triangle()	Конструктор класса треугольник
Triangle(std::istream &is)	Конструктор класса треугольник из стандартного потока ввода

Triangle(size_t i, size_t j, size_t k)	Конструктор класса треугольник по задаваемым в тексте программы значениям
Triangle(const Triangle& orig)	Конструктор копии класса треугольник
double Square()	Функция для вычисления площади фигур по её параметрам
void Print()	Функция для вывода фигуры на экран
~Triangle()	Деструктор класса треугольник
Rectangle.cpp	
Rectangle ()	Конструктор класса прямоугольник
Rectangle (std::istream &is)	Конструктор класса прямоугольник из стандартного потока ввода
Rectangle (size_t i, size_t j, size_t k)	Конструктор класса прямоугольник по задаваемым в тексте программы значениям
Rectangle (const Triangle& orig)	Конструктор копии класса прямоугольник
double Square()	Функция для вычисления площади фигур по её параметрам
void Print()	Функция для вывода фигуры на экран
~Rectangle ()	Деструктор класса прямоугольник
FSquare.cpp	
FSquare ()	Конструктор класса квадрат
FSquare (std::istream &is)	Конструктор класса квадрат из стандартного потока ввода
FSquare (size_t i, size_t j, size_t k)	Конструктор класса квадрат по задаваемым в тексте программы значениям
FSquare (const FSquare & orig)	Конструктор копии класса квадрат
double Square()	Функция для вычисления площади фигур по её параметрам
void Print()	Функция для вывода фигуры на экран
~ FSquare ()	Деструктор класса квадрат

Основные понятия ООП

Класс – это способ описания сущности, определяющий состояние и поведение, зависящее от этого состояния, а также правила для взаимодействия с данной сущностью (контракт).

С точки зрения программирования класс можно рассматривать как набор данных (полей, атрибутов, членов класса) и функций для работы с ними (методов).

Объект (экземпляр) – это отдельный представитель класса, имеющий конкретное состояние и поведение, полностью определяемое классом.

Интерфейс – это набор методов класса, доступных для использования другими классами.

Ключевые черты ООП

- **инкапсуляция** — это определение классов — пользовательских типов данных, объединяющих своё содержимое в единый тип и реализующих некоторые операции или методы над ним. Классы обычно являются основой модульности, инкапсуляции и абстракции данных в языках ООП.
- **наследование** — способ определения нового типа, когда новый тип наследует свойства и методы существующего, модифицируя или расширяя их.
- **полиморфизм**, позволяет единообразно ссылаться на объекты различных классов (обычно внутри некоторой иерархии).

конструктор класса — специальный блок инструкций, вызываемый при создании объекта.

Деструктор – специальный метод класса, служащий для удаления объекта

виртуальная функция — в объектно-ориентированном программировании метод (функция) класса, который может быть переопределён в классах-наследниках

Листинг программы:

Вывод:

При выполнении данной лабораторной работы я приобрел навыки написания классов в языке программирования C++. Классы являются множествами объектов со схожими свойствами, для их описания требуется задать параметры объектов и методы работы с ними. В данной лабораторной работе встречаются виртуальные методы, главной особенностью которых является наследование этих методов классами потомками, таким образом возможно описать их для класса-предка и не переопределяя его, учитывая атрибуты класса-потомка, определить метод работы с потомком.