

*Лабораторная работа 3*  
**КЛАССЫ: ПОЛЯ И МЕТОДЫ**

**Цель работы**

1. Изучить:
  - объявление класса в программе;
  - объявление и определение методов класса;
  - объявление различных объектов класса и вызов методов для них;
  - структуру программы с классами.
2. Составить, выполнить и протестировать указанную задачу с помощью компьютера.

**Порядок написания программы**

1. Внимательно прочитать условие задач.
2. Объявить класс в соответствии с условием задачи:
  - Объявить данные класса;
  - Объявить методы класса;
  - Определить методы класса.
3. Написать функцию main и объявить различные объекты:
  - Простую переменную статически;
  - Простую переменную динамически;
  - Массив объектов;
4. Вызвать каждый метод для каждого объекта.
5. Создать проект. Выполнить проект.

**Контрольные вопросы и задания**

1. Что такое класс?
2. Назовите компоненты класса.
3. Что такое методы? Чем они отличаются от обычных функций?
4. Какой метод называется константным?
5. Как регулируется доступ к отдельным компонентам класса?
6. В чем различие методов, определенных вне и внутри класса?
7. Что такое this?
8. Как объявить объект класса, статически, динамически?
9. Как, зная имя объекта, обратиться к полям и методам класса?
10. Как, зная адрес объекта, обратиться к полям и методам класса?
11. Для наилучшего освоения темы выполните варианты 1, 22, 24.

**Задание**

**Написать программу с использованием классов.**

### **Вариант 1**

Объявить и определить класс – точка на плоскости:

Данные:

- ✓ наименование объекта;
- ✓ координаты точки.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение расстояния между двумя точками.

В пользовательской функции main протестировать методы.

### **Вариант 2**

Объявить и определить класс – круг на плоскости:

Данные:

- ✓ наименование объекта;
- ✓ координаты центра и радиус.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение площади круга.

В пользовательской функции main протестировать методы.

### **Вариант 3**

Объявить и определить класс – прямоугольник на плоскости:

Данные:

- ✓ наименование объекта;
- ✓ координаты нижнего левого и верхнего правого угла прямоугольника (стороны параллельны осям).

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение площади прямоугольника.

В пользовательской функции main протестировать методы.

### **Вариант 4**

Объявить и определить класс – трапеция на плоскости:

Данные:

- ✓ наименование объекта;
- ✓ основания и высота.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение площади трапеции.

В пользовательской функции main протестировать методы.

### **Вариант 5**

Объявить и определить класс – правильный  $n$ -угольник на плоскости:

Данные:

- ✓ наименование объекта;
- ✓ количество сторон;
- ✓ длина одной стороны.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение периметра правильного  $n$ -угольника.

В пользовательской функции main протестировать методы.

### **Вариант 6**

Объявить и определить класс – треугольник на плоскости:

Данные:

- ✓ наименование объекта;
- ✓ стороны треугольника.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение площади треугольника.

В пользовательской функции main протестировать методы.

### **Вариант 7**

Объявить и определить класс – вектор на плоскости:

Данные:

- ✓ наименование объекта;
- ✓ координаты начала и конца вектора.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение длины вектора.

В пользовательской функции main протестировать методы.

### **Вариант 8**

Объявить и определить класс – точка в пространстве:

Данные:

- ✓ наименование объекта;
- ✓ координаты точки в пространстве.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение расстояния между двумя точками.

В пользовательской функции main протестировать методы.

### **Вариант 9**

Объявить и определить класс – круглый цилиндр:

Данные:

- ✓ наименование объекта;
- ✓ координаты центра окружности, радиус и высота.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение объема цилиндра.

В пользовательской функции main протестировать методы.

### **Вариант 10**

Объявить и определить класс – круглый конус:

Данные:

- ✓ наименование объекта;
- ✓ координаты центра окружности, радиус и высота.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение объема конуса.

В пользовательской функции main протестировать методы.

### **Вариант 11**

Объявить и определить класс – параллелепипед:

Данные:

- ✓ наименование объекта;
- ✓ стороны параллелепипеда.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение объема параллелепипеда.

В пользовательской функции main протестировать методы.

### **Вариант 12**

Объявить и определить класс - шар:

Данные:

- ✓ наименование объекта;
- ✓ координаты центра и радиус.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение объема шара.

В пользовательской функции main протестировать методы.

### **Вариант 13**

Объявить и определить класс – многочлен второй степени:

Данные:

- ✓ наименование объекта;
- ✓ коэффициенты многочлена.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение есть ли действительные корни.

В пользовательской функции main протестировать методы.

### **Вариант 14**

Объявить и определить класс – угол:

Данные:

- ✓ наименование объекта;
- ✓ количество градусов, минут и секунд.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ перевод значения угла в секунды;

В пользовательской функции main протестировать методы.

### **Вариант 15**

Объявить и определить класс – деньги:

Данные:

- ✓ наименование объекта;
- ✓ рубли и копейки.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение количества денег в копейках;

В пользовательской функции main протестировать методы.

### **Вариант 16**

Объявить и определить класс – погода:

Данные:

- ✓ наименование объекта;
- ✓ порядковый номер дня в году;
- ✓ температура;
- ✓ количество осадков.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ из двух объектов вычислить максимальную температуру.

В пользовательской функции main протестировать методы.

### **Вариант 17**

Объявить и определить класс – длина:

Данные:

- ✓ наименование объекта;
- ✓ количество метров, сантиметров, миллиметров.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ перевод длины в миллиметры;

В пользовательской функции main протестировать методы.

### **Вариант 18**

Объявить и определить класс – вес:

Данные:

- ✓ наименование объекта;
- ✓ количество килограмм и грамм.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ вычисление веса в граммах;

В пользовательской функции main протестировать методы.

### **Вариант 19**

Объявить и определить класс – возраст:

Данные:

- ✓ наименование объекта;
- ✓ количество лет, месяцев и дней.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ вычисление возраста в днях;

В пользовательской функции main протестировать методы.

### **Вариант 20**

Объявить и определить класс – кошка:

Данные:

- ✓ наименование объекта;
- ✓ вес и возраст.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора.

В пользовательской функции main протестировать методы.

### **Вариант 21**

Объявить и определить класс – собака:

Данные:

- ✓ наименование объекта;
- ✓ возраст и рост.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора.

В пользовательской функции main протестировать методы.

### **Вариант 22**

Объявить и определить класс – натуральная дробь:

Данные:

- ✓ наименование объекта;
- ✓ числитель и знаменатель.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ сокращение дроби.

В пользовательской функции main протестировать методы.

### **Вариант 23**

Объявить и определить класс – комплексное число:

Данные:

- ✓ наименование объекта;
- ✓ действительная и мнимая части.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ определение модуля комплексного числа.

В пользовательской функции main протестировать методы.

### **Вариант 24**

Объявить и определить класс – дата:

Данные:

- ✓ наименование объекта;
- ✓ день, месяц, год.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ превратить дату в число типа long, в котором сначала идут 4 цифры года, затем 2 цифры месяца и 2 цифры дня.

В пользовательской функции main протестировать методы.

## Вариант 25

Объявить и определить класс – время:

Данные:

- ✓ наименование объекта;
- ✓ часы, минуты и секунды.

Методы:

- ✓ заполнение данных с клавиатуры;
- ✓ вывод объекта на экран монитора;
- ✓ вычисление времени в секундах.

В пользовательской функции main протестировать методы.

### *Лабораторная работа 4*

## **КОНСТРУКТОРЫ И ДЕСТРУКТОРЫ КЛАССА**

### **Цель работы**

1. Изучить:
  - назначение конструкторов;
  - виды конструкторов;
  - особенности построения и вызова конструкторов;
  - назначение деструкторов;
  - правила построения деструкторов.
2. Составить, выполнить и протестировать указанную задачу с помощью компьютера.

### **Порядок написания программы**

1. Добавить в ранее созданный класс (лабораторная работа 3):
  - конструктор по умолчанию;
  - конструктор с параметрами;
  - конструктор копий;
  - деструктор.
2. Написать функцию main и создать объекты с помощью конструктора каждого вида. Вывести объекты на экран монитора.
5. Создать проект. Выполнить проект.

### **Контрольные вопросы и задания**

1. Что такое конструктор?
2. Какие действия в программе осуществляются с помощью конструктора?
3. Назовите виды конструкторов.
4. Какие особенности построения конструкторов, как функций?
5. Могут ли в классе отсутствовать конструкторы?
6. Когда конструктор должен быть в классе обязательно?



7. Может ли отсутствовать деструктор в классе? Когда он должен быть обязательно?

8. Для наилучшего освоения темы выполните варианты 1, 17, 20.

### **Задание**

Добавить в ранее созданный класс (лабораторная работа 3):

- конструктор по умолчанию, определяющий поля класса **случайным образом из диапазона возможных значений**;

- конструктор с параметрами, определяющий поля класса из параметров;

- конструктор копий;

- деструктор.

Написать функцию main и создать объекты с помощью конструктора каждого вида. Вывести объекты на экран монитора.

В последующих вариантах добавить еще один конструктор:

### **Вариант 14**

Класс – угол:

- ✓ конструктор с одним параметром – количество секунд.

### **Вариант 15**

Класс – деньги:

- ✓ конструктор с одним параметром – количество копеек.

### **Вариант 17**

Класс – длина:

- ✓ конструктор с одним параметром – количество миллиметров.

### **Вариант 18**

Класс – вес:

- ✓ конструктор с одним параметром – количество грамм.

### **Вариант 19**

Класс – возраст:

- ✓ конструктор с одним параметром – количество дней.

### **Вариант 25**

Класс – время:

- ✓ конструктор с одним параметром – количество секунд.

*Лабораторная работа 5*  
**ПЕРЕГРУЗКА ОПЕРАЦИЙ МЕТОДАМИ КЛАССА**

**Цель работы**

1. Изучить:
  - условия перегрузки операций методами класса;
  - синтаксис функции оператора;
  - правила перегрузки унарных операций;
  - правила перегрузки арифметических операций;
  - правила перегрузки бинарных операций.
2. Составить, выполнить и протестировать указанную задачу с помощью компьютера.

**Порядок написания программы**

1. Добавить в ранее созданный класс (лабораторные работы 3, 4) перегрузку методами класса указанных операций;  
Предварительно проанализировать характеристики операций, а именно:
  - количество операндов операции;
  - в чем суть операции, порядок ее выполнения;
  - появляются ли новые объекты в результате операции;
  - изменяются ли операнды в результате операции;
  - каков результат операции.
2. Написать функцию `main`, создать объекты и протестировать созданные методы.
5. Создать проект. Выполнить проект.

**Контрольные вопросы и задания**

1. Что такое функция оператор?
2. Можно ли менять семантику операций?
3. Можно ли менять количество операндов в операции?
4. Можно ли создавать новые операции?
5. Правила перегрузки унарных операций.
6. Правила перегрузки арифметических операций.
7. Правила перегрузки операции присваивания.
8. Когда перегрузка может осуществляться методами класса?

**Задание**

- Добавить в ранее созданный класс (лабораторные работы 3, 4) перегрузку методами класса указанных операций;
- Написать функцию `main`, создать объекты и протестировать созданные методы.

### **Вариант 1**

Класс – точка на плоскости:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект+вещественное число
- ✓ Объект1==Объект2

В пользовательской функции main протестировать методы.

### **Вариант 2**

Класс – круг на плоскости:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект+=вещественное число
- ✓ Объект1>Объект2

В пользовательской функции main протестировать методы.

### **Вариант 3**

Класс – прямоугольник на плоскости:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект\*=вещественное число
- ✓ Объект1>=Объект2

В пользовательской функции main протестировать методы.

### **Вариант 4**

Класс – трапеция на плоскости:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект++
- ✓ Объект1!=Объект2

В пользовательской функции main протестировать методы.

### **Вариант 5**

Класс – правильный n-угольник на плоскости:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект --
- ✓ Объект1 -=целое число

В пользовательской функции main протестировать методы.

### **Вариант 6**

Класс – треугольник на плоскости:

Переопределить методами класса операции:

- ✓ Объект1=Объект2

- ✓ Объект/вещественное число
- ✓ Объект1<Объект2

В пользовательской функции main протестировать методы.

### Вариант 7

Класс – вектор на плоскости:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект\*=вещественное число
- ✓ Объект1<=Объект2

В пользовательской функции main протестировать методы.

### Вариант 8

Класс – точка в пространстве:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1–Объект2
- ✓ ++Объект1

В пользовательской функции main протестировать методы.

### Вариант 9

Класс – круглый цилиндр:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект –вещественное число
- ✓ Объект1>= вещественное число

В пользовательской функции main протестировать методы.

### Вариант 10

Класс – круглый конус:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект/=вещественное число
- ✓ Объект1<=Объект2

В пользовательской функции main протестировать методы.

### Вариант 11

Класс – параллелепипед:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ – –Объект
- ✓ Объект1<Объект2

В пользовательской функции main протестировать методы.

### Вариант 12

Класс - шар:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект\*=вещественное число
- ✓ Объект1< вещественное число

В пользовательской функции main протестировать методы.

### **Вариант 13**

Класс – многочлен второй степени:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1/= вещественное число
- ✓ Объект1/ вещественное число

В пользовательской функции main протестировать методы.

### **Вариант 14**

Класс – угол:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1+Объект2
- ✓ Объект1>=Объект2

В пользовательской функции main протестировать методы.

### **Вариант 15**

Класс – деньги:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1+Объект2
- ✓ Объект1<=целое

В пользовательской функции main протестировать методы.

### **Вариант 16**

Класс – погода:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект– =вещественное число (или целое)
- ✓ Объект1!=Объект2

В пользовательской функции main протестировать методы.

### **Вариант 17**

Класс – длина:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1–Объект2
- ✓ Объект1>целое

В пользовательской функции main протестировать методы.

### **Вариант 18**

Класс – вес:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1+Объект2
- ✓ Объект1>Объект2

В пользовательской функции main протестировать методы.

### **Вариант 19**

Класс – возраст:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект\*=целое
- ✓ Объект1<Объект2

В пользовательской функции main протестировать методы.

### **Вариант 20**

Класс – кошка:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект++
- ✓ Объект1==Объект2

В пользовательской функции main протестировать методы.

### **Вариант 21**

Класс – собака:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ --Объект
- ✓ Объект1!=Объект2

В пользовательской функции main протестировать методы.

### **Вариант 22**

Класс – натуральная дробь:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1\*Объект2
- ✓ Объект1\*=Объект2

В пользовательской функции main протестировать методы.

### **Вариант 23**

Класс – комплексное число:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1-Объект2
- ✓ Объект1==вещественное число

В пользовательской функции main протестировать методы.

## Вариант 24

Класс – дата:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект1-Объект2
- ✓ Объект1>Объект2

В пользовательской функции main протестировать методы.

## Вариант 25

Класс – время:

Переопределить методами класса операции:

- ✓ Объект1=Объект2
- ✓ Объект+=целое
- ✓ Объект1!=целое

В пользовательской функции main протестировать методы.

### *Лабораторная работа 6*

## **ПЕРЕГРУЗКА ОПЕРАЦИЙ ДРУЖЕСТВЕННЫМИ ФУНКЦИЯМИ**

### **Цель работы**

1. Изучить:
  - условия перегрузки операций дружественными функциями;
  - синтаксис функции оператора для дружественных функций;
2. Составить, выполнить и протестировать указанную задачу с помощью компьютера.

### **Порядок написания программы**

1. Добавить в ранее созданный класс (лабораторные работы 3, 4, 5) перегрузку дружественными функциями указанных операций:
  - ввода из потока;
  - вывода в поток.
2. Написать функцию main, создать объекты и протестировать созданные методы.
5. Создать проект. Выполнить проект.

### **Контрольные вопросы и задания**

1. Когда применяются дружественные функции?
2. Провести сравнительный анализ перегрузки операций методами и дружественными функциями.
3. Принадлежат ли классу дружественные функции?
4. Имеют ли доступ к закрытым полям класса дружественные функции?
5. Передается ли функции один операнд по умолчанию? Левый или правый?

6. Обратитесь к выполненной лаб.раб. 3. Замените перегрузку любой операции дружественной функцией.

### **Задание**

Добавить в ранее созданный класс (лабораторные работы 3, 4, 5) перегрузку дружественными функциями указанных операций:

ввода из потока;

вывода в поток.

Написать функцию `main()`, создать объекты и протестировать созданные методы. Применить ввод-вывод к стандартным и **файловым** потокам.

### *Лабораторная работа 7*

## **СТАТИЧЕСКИЕ КОМПОНЕНТЫ КЛАССОВ**

### **Цель работы**

1. Изучить:

- назначение статических компонентов класса;
- синтаксис статических компонентов класса;

2. Составить, выполнить и протестировать указанную задачу с помощью компьютера.

### **Порядок написания программы**

1. В ранее созданный класс (лабораторные работы 3- 6) добавить статическое целое данное, контролирующее количество созданных объектов.

2. В функции `main` создавать объекты разными конструкторами и уничтожать их, каждый раз выводя на экран количество существующих объектов.

3. Создать проект. Выполнить проект.

### **Контрольные вопросы и задания**

1. Назначение статических компонентов класса.
2. Где и как объявляются статические компоненты класса.
3. Принадлежит ли статическое данное объектам класса?
4. Как вывести значение статической переменной класса?

### **Задание**

В ранее созданный класс (лабораторные работы 3- 6) добавить статическое целое данное, контролирующее количество созданных объектов. В функции `main` создавать объекты разными конструкторами и уничтожать их, каждый раз выводя на экран количество существующих объектов.



Создавайте и уничтожайте динамические объекты (это проще). На повышенную оценку создавать, *уничтожать* (и *контролировать*!) количество *статических* объектов.

**Используйте условную компиляцию для возможности отключения статического данного из класса.**