

Реализации всех классов, а также основная функция программы, должны быть разнесены по различным файлам исходного текста и использовать соответствующие заголовочные файлы.

## Абстрактные классы

В следующих заданиях требуется реализовать абстрактный базовый класс, определив в нем чистые виртуальные функции. Эти функции определяются в производных классах. В базовых классах должны быть объявлены чистые виртуальные функции ввода/вывода, которые реализуются в производных классах.

Вызывающая программа должна продемонстрировать все варианты вызова виртуальных функций с помощью указателей на базовый класс. Написать функцию вывода, получающую параметры базового класса по ссылке и демонстрирующую виртуальный вызов.

1. Создать абстрактный базовый класс **Figure** с виртуальными методами вычисления площади и периметра. Создать производные классы: **Rectangle** (прямоугольник), **Circle** (круг), **Trapezium** (трапеция) со своими функциями площади и периметра. Самостоятельно определить, какие поля необходимы, какие из них можно задать в базовом классе, а какие — в производных. Площадь трапеции:  $S = (a + b) \times h / 2$ .
2. Создать абстрактный базовый класс **Number** с виртуальными методами — арифметическими операциями. Создать производные классы **Integer** (целое) и **Real** (действительное).
3. Создать абстрактный базовый класс **Body** (тело) с виртуальными функциями вычисления площади поверхности и объема. Создать производные классы: **Parallelepiped** (параллелепипед) и **Ball** (шар) со своими функциями площади поверхности и объема.
4. Создать абстрактный класс **Currency** (валюта) для работы с денежными суммами. Определить виртуальные функции перевода в рубли и вывода на экран. Реализовать производные классы **Dollar** (доллар) и **Euro** (евро) со своими функциями перевода и вывода на экран.
5. Создать абстрактный базовый класс **Triangle** для представления треугольника с виртуальными функциями вычисления площади и периметра. Поля данных должны включать две стороны и угол между ними. Определить классы-наследники: прямоугольный треугольник, равнобедренный треугольник, равносторонний треугольник со своими функциями вычисления площади и периметра.
6. Создать абстрактный базовый класс **Pair** с виртуальными арифметическими операциями. Создать производные классы **FuzzyNumber** (задание 1.31) и **Complex** (задание 1.21).
7. Создать абстрактный базовый класс **Root** (корень) с виртуальными методами вычисления корней и вывода результата на экран. Определить производные классы **Linear** (линейное уравнение) и **Square** (квадратное уравнение) с собственными методами вычисления корней и вывода на экран.
8. Создать абстрактный базовый класс **Function** (функция) с виртуальными методами вычисления значения функции  $y = f(x)$  в заданной точке  $x$  и вывода результата на экран. Определить производные классы **Ellipse** (эллипс), **Hyperbola** (гипербола) с собственными функциями вычисления  $y$  в зависимости

от входного параметра  $x$ . Уравнение эллипса  $x^2 / a^2 + y^2 / b^2 = 1$ ; гиперболы:  $x^2 / a^2 - y^2 / b^2 = 1$ .

9. Создать абстрактный базовый класс `Pair` с виртуальными арифметическими операциями. Реализовать производные классы `Complex` (задание 1.21) и `Rational` (задание 1.28).
10. Создать абстрактный базовый класс `Triad` с виртуальными методами увеличения на 1. Создать производные классы `Date` (задание 1.29) и `Time` (задание 1.30).
11. Создать абстрактный базовый класс `Pair` с виртуальными арифметическими операциями. Создать производные классы `Money` (задание 1.24) и `Fraction` (задание 1.35).
12. Создать абстрактный базовый класс `Pair` с виртуальными арифметическими операциями. Реализовать производные классы `Money` (задание 1.24) и `Money` (задание 1.33).
13. Создать абстрактный базовый класс `Integer` (целое) с виртуальными арифметическими операциями и функцией вывода на экран. Определить производные классы `Decimal` (десятичное) и `Binary` (двоичное), реализующие собственные арифметические операции и функцию вывода на экран. Число представляется массивом, каждый элемент которого — цифра.
14. Создать абстрактный базовый класс `Pair` с виртуальными арифметическими операциями. Создать производные классы `Money` (задание 1.24) и `Complex` (задание 1.21).
15. Создать абстрактный базовый класс `Series` (прогрессия) с виртуальными функциями вычисления  $j$ -го элемента прогрессии и суммы прогрессии. Определить производные классы: `Linear` (арифметическая) и `Exponential` (геометрическая). (Арифметическая прогрессия  $a_j = a_0 + jd, j = 0, 1, 2, \dots$ . Сумма арифметической прогрессии:  $s_n = (n + 1) (a_0 + a_n) / 2$ . Геометрическая прогрессия:  $a_j = a_0 r^j, j = 0, 1, 2, \dots$ . Сумма геометрической прогрессии:  $s_n = (a_0 - a_n r) / (1 - r)$ .)
16. Создать абстрактный класс `Norm` с виртуальной функцией вычисления нормы и модуля. Определить производные классы `Complex`, `Vector3D` с собственными функциями вычисления нормы и модуля. (Модуль для комплексного числа вычисляется как корень из суммы квадратов действительной и мнимой частей; норма для комплексных чисел вычисляется как модуль в квадрате. Модуль вектора вычисляется как корень квадратный из суммы квадратов координат; норма вектора вычисляется как максимальное из абсолютных значений координат.)
17. Создать абстрактный базовый класс `Pair` с виртуальными арифметическими операциями. Создать производные классы `FuzzyNumber` (задание 1.31) и `Fraction` (задание 1.35).
18. Создать абстрактный базовый класс `Container` с виртуальными методами `sort()` и поэлементной обработки контейнера `foreach()`. Разработать производные классы `Bubble` (пузырек) и `Choice` (выбор). В первом классе сортировка реализуется методом пузырька, а поэлементная обработка состоит в извлечении

квадратного корня. Во втором классе сортировка реализуется методом выбора, а поэлементная обработка — вычисление логарифма.

19. Создать абстрактный базовый класс **Array** с виртуальными методами сложения и поэлементной обработки массива **foreach()**. Разработать производные классы **SortArray** и **XorArray**. В первом классе операция сложения реализуется как объединение множеств, а поэлементная обработка — сортировка. Во втором классе операция сложения реализуется как исключающее ИЛИ, а поэлементная обработка — вычисление корня.
20. Создать абстрактный базовый класс **Array** с виртуальными методами сложения и поэлементной обработки массива **foreach()**. Разработать производные классы **AndArray** и **OrArray** (выбор). В первом классе операция сложения реализуется как пересечение множеств, а поэлементная обработка представляет собой извлечение квадратного корня. Во втором классе операция сложения реализуется как объединение, а поэлементная обработка — вычисление логарифма.