**Варианты:**

Вариант 1

Создать абстрактный класс Издание с виртуальными методами позволяющим вывести на экран информацию об издании, а также определить является ли данное издание искомым.

Создать производные классы: Книга (название, фамилия автора, год издания, издательство), Статья (название, фамилия автора, название журнала, его номер и год издания), Электронный ресурс (название, фамилия автора, ссылка, аннотация) со своими методами вывода информации на экран.

Создать каталог (массив) из n изданий, вывести полную информацию из каталога, а также организовать поиск изданий по фамилии автора.

### Вариант 2

Создать абстрактный класс Point (точка). На его основе создать классы ColoredPoint и Line. На основе класса Line создать класс ColoredLine и класс PolyLine (многоугольник). Все классы должны иметь виртуальные методы установки и получения значений всех координат, а также изменения цвета и получения текущего цвета.

Создать класс Picture, содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 3

Создать абстрактный класс Vehicle. На его основе реализовать классы Car (автомобиль), Bicycle (велосипед) и Lorry (грузовик). Классы должны иметь возможность задавать и получать параметры средств передвижения (цена, максимальная скорость, год выпуска и т.д.). Наряду с общими полями и методами, каждый класс должен содержать и специфичные для него поля.

Создать класс Garage, содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 4

Создать абстрактный класс Figure. На его основе реализовать классы Rectangle (прямоугольник), Circle (круг) и Trapezium (трапеция) с возможностью вычисления площади, центра тяжести и периметра.

Создать класс Picture, содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 5

Создать абстрактный класс Number c виртуальными методами, реализующими арифметические операции. На его основе реализовать классы Integer и Real.

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 6

Создать абстрактный класс Body. На его основе реализовать классы Parallelepiped (прямоугольный параллелепипед), Cone (конус) и Ball (шар) с возможностью вычисления площади поверхности и объема.

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 7

Создать абстрактный класс Currency для работы с денежными суммами. Определить в нем методы перевода в рубли и вывода на экран. На его основе реализовать классы Dollar, Euro и Pound (фунт стерлингов) с возможностью пересчета в центы и пенсы соответственно.

Создать класс Purse (кошелек), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода общей суммы, переведенной в рубли, и суммы по каждой из валют. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 8

Создать абстрактный класс Triangle (треугольник), задав в нем длину двух сторон, угол между ними, методы вычисления площади и периметра. На его основе создать классы, описывающие равносторонний, равнобедренный и прямоугольный треугольники со своими методами вычисления площади и периметра.

Создать класс Picture, содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка и получения суммарной площади. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 9

Создать абстрактный класс Solution (решение) с виртуальными методами вычисления корней уравнения и вывода на экран. На его основе реализовать классы Linear (линейное уравнение) и Square (квадратное уравнение).

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 10

Создать абстрактный класс Function (функция) с виртуальными методами вычисления значения функции y = f(x) в заданной точке х и вывода результата на экран. На его основе реализовать классы Ellipse, Hiperbola и Parabola.

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 11

Создать абстрактный класс Triad (тройка) с виртуальными методами увеличения на 1. На его основе реализовать классы Date (дата) и Time (время).

Создать класс Memories, содержащий массив/параметризованную коллекцию пар (дата-время) объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка и выборки самого раннего и самого позднего событий. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 12

Описать абстрактный класс Element (элемент логической схемы), задав в нем числовой идентификатор, количество входов, идентификаторы присоединенных к нему элементов (до 10) и двоичные значения на входах и выходе. На его основе реализовать классы AND и OR — двоичные вентили, которые могут иметь различное количество входов и один выход и реализуют логическое умножение и сложение соответственно.

Создать класс Sсheme (схема), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможности вывода характеристик объектов списка и вычисление значений, формируемых на выходах схемы по заданным значениям входов. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 13

Описать абстрактный класс Element (элемент логической схемы) задав в нем символьный идентификатор, количество входов, идентификаторы присоединенных к нему элементов (до 10) и двоичные значения на входах и выходе. На его основе реализовать классы AND\_NOT и OR\_NOT — двоичные вентили, которые могут иметь различное количество входов и один выход и реализуют логическое умножение c отрицанием и сложение c отрицанием соответственно.

Создать класс Sсheme (схема), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможности вывода характеристик объектов списка и вычисление значений, формируемых на выходах схемы по заданным значениям входов. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 14

Описать абстрактный класс Trigger (триггер), задав в нем идентификатор и двоичные значения на входах и выходах. На его основе реализовать классы RS и JK, представляющие собой триггеры соответствующего типа.

Создать класс Register (регистр), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможности вывода характеристик объектов списка, общего сброса и установки значений каждого триггера по заданным значениям входов. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 15

Создать абстрактный класс Progression (прогрессия) с виртуальными методами вычисления заданного элемента и суммы прогрессии. На его основе реализовать классы Linear (арифметическая) и Exponential (геометрическая).

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка и вывода общей суммы всех прогрессий. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 16

Создать абстрактный класс Pair (пара значений) с виртуальными методами, реализующими арифметические операции. На его основе реализовать классы Fractional (дробное) и LongLong (длинное целое).

В классе Fractional вещественное число представляется в виде двух целых, в которых хранятся целая и дробная часть числа соответственно. В классе LongLong длинное целое число хранится в двух целых полях в виде старшей и младшей части.

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода характеристик объектов списка и вывода общей суммы всех значений. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 17

Создать абстрактный класс Integer (целое) с символьным идентификатором, виртуальными методами, реализующими арифметические операции, и методом вывода на экран. На его основе реализовать классы Decimal (десятичное) и Binary (двоичное). Число представить в виде массива цифр.

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода значений и идентификаторов всех объектов списка и вывода общей суммы всех десятичных значений. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 18

Создать абстрактный класс Sorting (сортировка) с идентификатором последовательности, виртуальными методами сортировки, получения суммы и вывода на экран. На его основе реализовать классы Choice (метод выбора) и Quick (быстрая сортировка).

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода идентификаторов и сумм элементов каждого объекта списка, а также вывода общей суммы всех значений. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 19

Создать абстрактный класс Pair (пара значений) с виртуальными методами, реализующими арифметические операции, и методом вывода на экран. На его основе реализовать классы Money (деньги) и Complex (комплексное число).

В классе Money денежная сумма представляется в виде двух целых, в которых хранятся рубли и копейки соответственно. При выводе части числа снабжаются словами «руб.» и «коп.». В классе Complex предусмотреть при выводе символ мнимой части (i).

Создать класс Series (набор), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода объектов списка. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Вариант 20

Создать абстрактный класс Worker с полями, задающими фамилию работника, фамилии руководителя и подчиненных и виртуальными методами вывода списка обязанностей и списка подчиненных на экран. На его основе реализовать классы Manager (руководитель проекта), Developer (разработчик) и Coder (младший программист).

Создать класс Group (группа), содержащий массив/параметризованную коллекцию объектов этих классов в динамической памяти. Предусмотреть возможность вывода всех объектов списка и выборки по фамилии с выводом всего дерева подчиненных. Написать демонстрационную программу, в которой будут использоваться все методы классов.

Дополнительное задание: дополнить класс методами сортировки по некоторому критерию, вывода в файл и считывания из файла.

### Задание 21

Создать абстрактный класс Persona с методами, позволяющим вывести на экран информацию о персоне, а также определить ее возраст (на момент текущей даты).

Создать производные классы: Абитуриент (фамилия, дата рождения, факультет), Студент (фамилия, дата рождения, факультет, курс), Преподавать (фамилия, дата рождения, факультет, должность, стаж), со своими методами вывода информации на экран, и определения возраста.

Создать базу (массив) из n персон, вывести полную информацию из базы на экран, а также организовать поиск персон, чей возраст попадает в заданный диапазон.

### Задание 22

Создать абстрактный класс Клиент с методами, позволяющими вывести на экран информацию о клиентах банка, а также определить соответствие клиента критерию поиска.

Создать производные классы: Вкладчик (фамилия, дата открытия вклада, размер вклада, процент по вкладу), Кредитор (фамилия, дата выдачи кредита, размер кредита, процент по кредиту, остаток долга), Организация (название, дата открытия счета, номер счета, сумма на счету) со своими методами вывода информации на экран, и определения соответствия дате (открытия вклада, выдаче кредита, открытия счета).

Создать базу (массив) из n клиентов, вывести полную информацию из базы на экран, а также организовать поиск клиентов, начавших сотрудничать с банком в заданную дату.

Задание 23

Создать абстрактный класс Товар с методами, позволяющими вывести на экран информацию о товаре, а также определить, соответствует ли она искомому типу.

Создать производные классы: Игрушка (название, цена, производитель, материал, возраст, на который рассчитана), Книга (название, автор, цена, издательство, возраст, на который рассчитана), Спорт-инвентарь (название, цена, производитель, возраст, на который рассчитана), со своими методами вывода информации на экран, и определения соответствия искомому типу.

Создать базу (массив) из n товаров, вывести полную информацию из базы на экран, а также организовать поиск товаров определенного типа.