**Тема: «Методы. Реализация полиморфизма»**

**Задание 1. Работа со статическими методами**

1. Создайте статический метод, который вычисляет факториал числа.
2. Напишите статический метод, который возвращает сумму элементов массива.
3. Создайте статический метод, который проверяет, является ли строка палиндромом.
4. Реализуйте статический метод, который находит наибольший общий делитель двух чисел.
5. Напишите статический метод, который преобразует число в двоичное представление.
6. Создайте статический метод, который сортирует массив чисел по возрастанию.
7. Реализуйте статический метод, который вычисляет среднее значение массива.
8. Напишите статический метод, который проверяет, является ли число простым.
9. Создайте статический метод, который возвращает количество слов в строке.
10. Реализуйте статический метод, который находит индекс элемента в массиве.
11. Напишите статический метод, который переворачивает строку.
12. Создайте статический метод, который вычисляет площадь круга по радиусу.
13. Реализуйте статический метод, который проверяет, является ли год високосным.
14. Напишите статический метод, который возвращает массив уникальных элементов из входного массива.
15. Создайте статический метод, который вычисляет сумму цифр числа.

**Задание 2**. Методы не возвращающие значения (процедуры). Способы передачи параметров в методы.

* + 1. Описать процедуру PowerA3(A, B), вычисляющую третью степень числа A и возвращающую ее в переменной B (A — входной, B — выходной параметр; оба параметра являются вещественными). С помощью этой процедуры найти третьи степени пяти данных чисел.
    2. Описать процедуру PowerA234(A, B, C, D), вычисляющую вторую, третью и четвертую степень числа A и возвращающую эти степени соответственно в переменных B, C и D (A — входной, B, C, D — выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти вторую, третью и четвертую степень пяти данных чисел.
    3. Описать процедуру TrianglePS(a, P, S), вычисляющую по стороне a равностороннего треугольника его периметр P = 3·a и площадь S =√3/4∙a^2 (a — входной, P и S — выходные параметры; все параметры являются вещественными). С помощью этой процедуры найти периметры и площади трех равносторонних треугольников с данными сторонами.
    4. Описать процедуру RectPS(x1, y1, x2, y2, P, S), вычисляющую периметр P и площадь S прямоугольника со сторонами, параллельными осям координат, по координатам (x1, y1), (x2, y2) его противоположных вершин (x1, y1, x2, y2 — входные, P и S — выходные параметры вещественного типа). С помощью этой процедуры найти периметры и площади трех прямоугольников с данными противоположными вершинами.
    5. Описать процедуру DigitCountSum(K, C, S), находящую количество C цифр целого положительного числа K, а также их сумму S (K — входной, C и S — выходные параметры целого типа). С помощью этой процедуры найти количество и сумму цифр для каждого из пяти данных целых чисел.
    6. Описать процедуру InvDigits(K), меняющую порядок следования цифр целого положительного числа K на обратный (K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры поменять порядок следования цифр на обратный для каждого из пяти данных целых чисел.
    7. Описать процедуру AddRightDigit(D, K), добавляющую к целому положительному числу K справа цифру D (D — входной параметр целого типа, лежащий в диапазоне 0–9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры последовательно добавить к данному числу K справа данные цифры D1 и D2, выводя результат каждого добавления.
    8. Описать процедуру AddLeftDigit(D, K), добавляющую к целому положительному числу K слева цифру D (D — входной параметр целого типа, лежащий в диапазоне 1–9, K — параметр целого типа, являющийся одновременно входным и выходным). С помощью этой процедуры последовательно добавить к данному числу K слева данные цифры D1 и D2, выводя результат каждого добавления.
    9. Описать процедуру Swap(X, Y), меняющую содержимое переменных X и Y (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). С ее помощью для данных переменных A, B, C, D последовательно поменять содержимое следующих пар: A и B, C и D, B и C и вывести новые значения A, B, C, D.
    10. Описать процедуру Minmax(X, Y), записывающую в переменную X минимальное из значений X и Y, а в переменную Y — максимальное из этих значений (X и Y — вещественные параметры, являющиеся одновременно входными и выходными). Используя четыре вызова этой процедуры, найти минимальное и максимальное из данных чисел A, B, C, D.
    11. Описать процедуру SortInc3(A, B, C), меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по возрастанию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры упорядочить по возрастанию два данных набора из трех чисел: (A1, B1, C1) и (A2, B2, C2).
    12. Описать процедуру SortDec3(A, B, C), меняющую содержимое переменных A, B, C таким образом, чтобы их значения оказались упорядоченными по убыванию (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры упорядочить по убыванию два данных набора из трех чисел: (A1, B1, C1) и (A2, B2, C2).
    13. Описать процедуру ShiftRight3(A, B, C), выполняющую правый циклический сдвиг: значение A переходит в B, значение B — в C, значение C — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры выполнить правый циклический сдвиг для двух данных наборов из трех чисел: (A1, B1, C1) и (A2, B2, C2).
    14. Описать процедуру ShiftLeft3(A, B, C), выполняющую левый циклический сдвиг: значение A переходит в C, значение C — в B, значение B — в A (A, B, C — вещественные параметры, являющиеся одновременно входными и выходными). С помощью этой процедуры выполнить левый циклический сдвиг для двух данных наборов из трех чисел: (A1, B1, C1) и (A2, B2, C2).
    15. Описать процедуру Mean(X, Y, AMean, GMean), вычисляющую среднее арифметическое AMean = (X + Y)/2 и среднее геометрическое GMean = X ⋅Y двух положительных чисел X и Y (X и Y — входные, AMean и GMean — выходные параметры вещественного типа). С помощью этой процедуры найти среднее арифметическое и среднее геометрическое для пар (A, B), (A, C), (A, D), если даны A, B, C, D.

**Задание 3. Работа с методами и рекурсией (уровень сложности 4)**

Это задание направлено на углубленное понимание методов, включая рекурсию, передачу параметров по значению и ссылке, а также работу с исключениями. Каждая задача требует написания метода для решения конкретной проблемы.

**Задачи**

1. **Напишите метод, который вычисляет факториал числа с использованием рекурсии.**
   * Метод должен принимать целое число и возвращать его факториал.
   * Пример: Factorial(5) → 120.
2. **Создайте метод, который вычисляет числа Фибоначчи с использованием рекурсии.**
   * Метод должен принимать номер числа Фибоначчи и возвращать его значение.
   * Пример: Fibonacci(6) → 8.
3. **Напишите метод, который проверяет, является ли строка палиндромом, используя рекурсию.**
   * Метод должен возвращать true, если строка является палиндромом, и false в противном случае.
   * Пример: IsPalindrome("racecar") → true.
4. **Создайте метод, который вычисляет наибольший общий делитель (НОД) двух чисел с использованием рекурсии.**
   * Метод должен использовать алгоритм Евклида.
   * Пример: GCD(56, 98) → 14.
5. **Напишите метод, который реверсирует строку с использованием рекурсии.**
   * Метод должен возвращать перевернутую строку.
   * Пример: ReverseString("hello") → "olleh".
6. **Создайте метод, который вычисляет сумму цифр числа с использованием рекурсии.**
   * Метод должен принимать целое число и возвращать сумму его цифр.
   * Пример: SumOfDigits(1234) → 10.
7. **Напишите метод, который проверяет, является ли число простым, используя рекурсию.**
   * Метод должен возвращать true, если число простое, и false в противном случае.
   * Пример: IsPrime(17) → true.
8. **Создайте метод, который вычисляет степень числа с использованием рекурсии.**
   * Метод должен принимать два числа: основание и показатель степени.
   * Пример: Power(2, 5) → 32.
9. **Напишите метод, который находит минимальный элемент в массиве с использованием рекурсии.**
   * Метод должен принимать массив целых чисел и возвращать минимальное значение.
   * Пример: FindMin(new int[] { 5, 3, 9, 1 }) → 1.
10. **Создайте метод, который вычисляет сумму элементов массива с использованием рекурсии.**
    * Метод должен принимать массив целых чисел и возвращать сумму его элементов.
    * Пример: ArraySum(new int[] { 1, 2, 3, 4 }) → 10.
11. **Напишите метод, который проверяет, отсортирован ли массив по возрастанию, используя рекурсию.**
    * Метод должен возвращать true, если массив отсортирован, и false в противном случае.
    * Пример: IsSorted(new int[] { 1, 2, 3, 4 }) → true.
12. **Создайте метод, который находит индекс элемента в массиве с использованием рекурсии.**
    * Метод должен принимать массив и значение для поиска, возвращать индекс элемента или -1, если элемент не найден.
    * Пример: FindIndex(new int[] { 5, 3, 9, 1 }, 9) → 2.
13. **Напишите метод, который вычисляет количество вхождений элемента в массиве с использованием рекурсии.**
    * Метод должен принимать массив и значение для поиска, возвращать количество вхождений.
    * Пример: CountOccurrences(new int[] { 1, 2, 3, 2, 4, 2 }, 2) → 3.
14. **Создайте метод, который генерирует все перестановки строки с использованием рекурсии.**
    * Метод должен принимать строку и выводить все возможные перестановки.
    * Пример: Permute("abc") → ["abc", "acb", "bac", "bca", "cab", "cba"].
15. **Напишите метод, который решает задачу Ханойской башни с использованием рекурсии.**
    * Метод должен принимать количество дисков и выводить последовательность шагов для перемещения дисков.
    * Пример: HanoiTower(3, 'A', 'C', 'B') → Вывод последовательности шагов.

**Задание 4. Работа с расширяющими методами**

1. Создайте расширяющий метод для типа string, который возвращает количество гласных букв.
2. Напишите расширяющий метод для типа int, который проверяет, является ли число четным.
3. Реализуйте расширяющий метод для массива int[], который возвращает сумму элементов.
4. Создайте расширяющий метод для типа string, который удаляет все пробелы.
5. Напишите расширяющий метод для типа DateTime, который возвращает возраст в годах.
6. Реализуйте расширяющий метод для типа List<int>, который возвращает медиану.
7. Создайте расширяющий метод для типа string, который проверяет, содержит ли строка только цифры.
8. Напишите расширяющий метод для типа double, который округляет число до указанного количества знаков.
9. Реализуйте расширяющий метод для типа string, который возвращает первую букву каждого слова.
10. Создайте расширяющий метод для типа int[], который возвращает массив уникальных элементов.
11. Напишите расширяющий метод для типа string, который возвращает строку в обратном порядке.
12. Реализуйте расширяющий метод для типа List<string>, который сортирует строки по длине.
13. Создайте расширяющий метод для типа int, который возвращает факториал числа.
14. Напишите расширяющий метод для типа string, который заменяет все гласные на звездочки.
15. Реализуйте расширяющий метод для типа DateTime, который возвращает день недели на русском языке.

**Задание 5. Работа с абстрактными классами, виртуальными и абстрактными методами (уровень сложности 4)**

Это задание направлено на углубленное понимание абстрактных классов, виртуальных и абстрактных методов. В каждой задаче требуется создать абстрактный класс с виртуальными и абстрактными методами, а также два класса-наследника, которые реализуют или переопределяют эти методы.

#### **Задачи**

1. **Создайте абстрактный класс Shape с абстрактным методом CalculateArea() и виртуальным методом DisplayInfo().**
   * Реализуйте два класса-наследника: Circle и Rectangle.
   * В классе Circle реализуйте метод CalculateArea() для вычисления площади круга.
   * В классе Rectangle реализуйте метод CalculateArea() для вычисления площади прямоугольника.
   * Переопределите метод DisplayInfo() в обоих классах для вывода информации о фигуре.
2. **Создайте абстрактный класс Animal с абстрактным методом MakeSound() и виртуальным методом Sleep().**
   * Реализуйте два класса-наследника: Dog и Cat.
   * В классе Dog реализуйте метод MakeSound() для вывода "Woof!".
   * В классе Cat реализуйте метод MakeSound() для вывода "Meow!".
   * Переопределите метод Sleep() в обоих классах для вывода сообщения о том, как спит животное.
3. **Создайте абстрактный класс Vehicle с абстрактным методом Move() и виртуальным методом Stop().**
   * Реализуйте два класса-наследника: Car и Bicycle.
   * В классе Car реализуйте метод Move() для вывода "Car is moving".
   * В классе Bicycle реализуйте метод Move() для вывода "Bicycle is moving".
   * Переопределите метод Stop() в обоих классах для вывода сообщения о том, как транспортное средство останавливается.
4. **Создайте абстрактный класс Employee с абстрактным методом CalculateSalary() и виртуальным методом DisplayInfo().**
   * Реализуйте два класса-наследника: Manager и Developer.
   * В классе Manager реализуйте метод CalculateSalary() для возврата фиксированной зарплаты.
   * В классе Developer реализуйте метод CalculateSalary() для расчета зарплаты на основе часов работы.
   * Переопределите метод DisplayInfo() в обоих классах для вывода информации о сотруднике.
5. **Создайте абстрактный класс Device с абстрактным методом TurnOn() и виртуальным методом TurnOff().**
   * Реализуйте два класса-наследника: Smartphone и Laptop.
   * В классе Smartphone реализуйте метод TurnOn() для вывода "Smartphone is turning on".
   * В классе Laptop реализуйте метод TurnOn() для вывода "Laptop is turning on".
   * Переопределите метод TurnOff() в обоих классах для вывода сообщения о выключении устройства.
6. **Создайте абстрактный класс BankAccount с абстрактным методом CalculateInterest() и виртуальным методом DisplayBalance().**
   * Реализуйте два класса-наследника: SavingsAccount и CurrentAccount.
   * В классе SavingsAccount реализуйте метод CalculateInterest() для расчета процентов по сберегательному счету.
   * В классе CurrentAccount реализуйте метод CalculateInterest() для расчета процентов по текущему счету.
   * Переопределите метод DisplayBalance() в обоих классах для вывода текущего баланса.
7. **Создайте абстрактный класс Shape3D с абстрактным методом CalculateVolume() и виртуальным методом DisplayInfo().**
   * Реализуйте два класса-наследника: Sphere и Cube.
   * В классе Sphere реализуйте метод CalculateVolume() для вычисления объема сферы.
   * В классе Cube реализуйте метод CalculateVolume() для вычисления объема куба.
   * Переопределите метод DisplayInfo() в обоих классах для вывода информации о трехмерной фигуре.
8. **Создайте абстрактный класс MusicalInstrument с абстрактным методом PlaySound() и виртуальным методом Tune().**
   * Реализуйте два класса-наследника: Guitar и Piano.
   * В классе Guitar реализуйте метод PlaySound() для вывода "Guitar is playing".
   * В классе Piano реализуйте метод PlaySound() для вывода "Piano is playing".
   * Переопределите метод Tune() в обоих классах для вывода сообщения о настройке инструмента.
9. **Создайте абстрактный класс Food с абстрактным методом Cook() и виртуальным методом Serve().**
   * Реализуйте два класса-наследника: Pizza и Pasta.
   * В классе Pizza реализуйте метод Cook() для вывода "Pizza is cooking".
   * В классе Pasta реализуйте метод Cook() для вывода "Pasta is cooking".
   * Переопределите метод Serve() в обоих классах для вывода сообщения о подаче блюда.
10. **Создайте абстрактный класс Book с абстрактным методом Read() и виртуальным методом DisplayInfo().**
    * Реализуйте два класса-наследника: Novel и Textbook.
    * В классе Novel реализуйте метод Read() для вывода "Reading a novel".
    * В классе Textbook реализуйте метод Read() для вывода "Reading a textbook".
    * Переопределите метод DisplayInfo() в обоих классах для вывода информации о книге.
11. **Создайте абстрактный класс Computer с абстрактным методом Start() и виртуальным методом Shutdown().**
    * Реализуйте два класса-наследника: Desktop и Laptop.
    * В классе Desktop реализуйте метод Start() для вывода "Desktop is starting".
    * В классе Laptop реализуйте метод Start() для вывода "Laptop is starting".
    * Переопределите метод Shutdown() в обоих классах для вывода сообщения о выключении компьютера.
12. **Создайте абстрактный класс Plant с абстрактным методом Grow() и виртуальным методом DisplayInfo().**
    * Реализуйте два класса-наследника: Tree и Flower.
    * В классе Tree реализуйте метод Grow() для вывода "Tree is growing".
    * В классе Flower реализуйте метод Grow() для вывода "Flower is growing".
    * Переопределите метод DisplayInfo() в обоих классах для вывода информации о растении.
13. **Создайте абстрактный класс Game с абстрактным методом Play() и виртуальным методом DisplayRules().**
    * Реализуйте два класса-наследника: Chess и Checkers.
    * В классе Chess реализуйте метод Play() для вывода "Playing chess".
    * В классе Checkers реализуйте метод Play() для вывода "Playing checkers".
    * Переопределите метод DisplayRules() в обоих классах для вывода правил игры.
14. **Создайте абстрактный класс ElectronicDevice с абстрактным методом TurnOn() и виртуальным методом TurnOff().**
    * Реализуйте два класса-наследника: TV и Radio.
    * В классе TV реализуйте метод TurnOn() для вывода "TV is turning on".
    * В классе Radio реализуйте метод TurnOn() для вывода "Radio is turning on".
    * Переопределите метод TurnOff() в обоих классах для вывода сообщения о выключении устройства.
15. **Создайте абстрактный класс Sport с абстрактным методом Play() и виртуальным методом DisplayRules().**
    * Реализуйте два класса-наследника: Football и Basketball.
    * В классе Football реализуйте метод Play() для вывода "Playing football".
    * В классе Basketball реализуйте метод Play() для вывода "Playing basketball".
    * Переопределите метод DisplayRules() в обоих классах для вывода правил игры.