# Контрольная работа №5

Темы 1-4: “Классы и объекты. Свойства и методы класса. Конструктор, инициализатор, деструктор”

1. Создать класс **“Account”**, представляющий собой банковский счет. Класс должен содержать:
   1. динамические свойства: фамилия владельца, номер счета, процент начисления, сумма в рублях;
   2. статические свойства: курс рубля по отношению к доллару, курс рубля по отношению к евро;
   3. классовые методы: редактировать курс рубля по отношению к доллару, редактировать курс рубля по отношению к евро;
   4. статические методы: перевод суммы в доллары и евро;
   5. конструктор: вызывает конструктор родительского класса и выводит сообщение о создании нового банковского счета;
   6. инициализатор: определяет динамические свойства класса и выводит информацию об открытом счете;
   7. деструктор: выводит сообщение о том, что банковский счет закрыт;
   8. методы: смена владельца счета, снятие заданной суммы, начисление заданной суммы, начисление процентов, перевод в доллары и евро (в отличие от аналогичных статических методов, данные методы не принимают параметров), вывод информации о счете;

Продемонстрировать работу класса и всех его методов.

1. Создать класс **“Point”** для работы с точками на плоскости. Класс должен содержать:
   1. динамические свойства: координаты точки *X* и *Y*;
   2. статические свойства: количество точек, лежащих на оси *X*, количество точек, лежащих на оси *Y*, количество точек, совпадающих с началом координат;
   3. классовые методы: увеличить на 1 количество точек, лежащих на оси *X*, увеличить на 1 количество точек, лежащих на оси *Y*, увеличить на 1 количество точек, совпадающих с началом координат;
   4. статические методы: проверки, лежит ли точка на одной из осей координат или совпадает с началом координат;
   5. конструктор: вызывает конструктор родительского класса и выводит сообщение о создании новой точки;
   6. инициализатор: определяет динамические свойства класса и выводит созданный объект на экран;
   7. деструктор: выводит сообщение о том, что точка удалена;
   8. методы: перемещение точки по оси *X*, перемещение точки по оси *Y*, определение расстояния до начала координат, вычисление расстояния до заданной точки, сравнение на совпадение и несовпадение с заданной точкой, вывод точки на экран;

*\* заданная точка - также экземпляр класса “Point”.*

Продемонстрировать работу класса и всех его методов.

1. Создать класс **“Time”** для работы со временем в формате *Час:Минута:Секунда*. Класс должен содержать:
   1. динамические свойства: количество часов, минут, секунд;
   2. статическое свойство: часовой пояс (строка в формате “UTC+/-число”);
   3. классовые методы: редактировать часовой пояс;
   4. статические методы: проверка корректности заданных величин (часов, минут, секунд), перевод заданного значения из формата *Час:Минута:Секунда* в секунды и наоборот;
   5. конструктор: вызывает конструктор родительского класса и выводит сообщение о создании нового момента времени;
   6. инициализатор: проверяет корректность переданных величин, определяет динамические свойства класса и выводит на экран информацию об объекте;
   7. деструктор: выводит сообщение о том, что момент времени удален;
   8. методы: вычисление разницы между двумя моментами времени в секундах, сложение с заданным количеством секунд, вычитание заданного количества секунд, сравнение двух моментов времени, вывод на экран;

Продемонстрировать работу класса и всех его методов.

Тема 5: “Инкапсуляция. Модификаторы доступа”

Внести изменения в созданные классы следующим образом:

* 1. динамические свойства класса **“Account”** должны иметь модификатор доступа *private*;
  2. динамические свойства классов **“Point”** и **“Time”** должны иметь модификатор доступа *protected*;
  3. определить методы *getter* и *setter* для свойств с модификаторами доступа *private* и *protected*;
  4. любое взаимодействие с *private* и *protected* свойствами должно производиться посредством соответствующих методов *getter* и *setter*.

Продемонстрировать работу измененных классов.

Темы 6-7: “Наследование. Множественное наследование. Полиморфизм”

1. Создать класс **“Liquid”** (жидкость) со свойствами: название и плотность жидкости, - и методами: изменение плотности, вычисление объема жидкости, соответствующего заданной массе, вычисление массы жидкости, соответствующей заданному объему, вывод информации о жидкости. Создать производный класс **“Alcohol”** (спирт) с собственным свойством - крепость, - и методом: изменение крепости. Продемонстрировать работу класса-наследника и всех его методов.
2. Создать класс **“Pair”** (пара чисел) со свойствами: числа *A* и *B*, - и методами: изменение чисел, вычисление их произведения и суммы. Определить производный класс **“Right Triangle”** (прямоугольный треугольник) со свойствами: катеты *A* и *B*, - и методами: вычисление гипотенузы и площади треугольника, вывод информации о фигуре на экран. Продемонстрировать работу класса-наследника и всех его методов.
3. Внести изменения в созданный в первой задаче класс **“Alcohol”**: переопределить методы вычисления массы и объема жидкости таким образом, чтобы в них также рассчитывалось соответственно массовое или объемное содержание чистого спирта, исходя из заданной крепости. Переопределить метод вывода информации о спирте.

Тема 8: “Абстракция”

1. Создать базовый абстрактный класс **“Pair”** (пара чисел) с арифметическими операциями: сложение, вычитание, умножение и деление. Определить производный класс **“Fuzzy numbers”** (нечеткие числа), реализующий арифметические операции следующим образом:
   1. *A(A-a, A, A+a)* ***+*** *B(B-b, B, B+b) = ((A-a)+(B-b), A+B, (A+a)+(B+b))*
   2. *A(A-a, A, A+a)* ***-*** *B(B-b, B, B+b) = ((A-a)-(B-b), A-B, (A+a)-(B+b))*
   3. *A(A-a, A, A+a)* ***\**** *B(B-b, B, B+b) = ((A-a)\*(B-b), A\*B, (A+a)\*(B+b))*
   4. *A(A-a, A, A+a)* ***/*** *B(B-b, B, B+b) = ((A-a)/(B-b), A/B, (A+a)/(B+b))*

*\* решите самостоятельно, какими свойствами будет обладать каждый из классов, и какие методы следует определить как абстрактные.*

1. Создать базовый абстрактный класс **“Array”** (массив) с методами сложения и поэлементной обработки. Определить производные классы **“AndArray”** и **“OrArray”**. 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.

*\* решите самостоятельно, какими свойствами будет обладать каждый из классов, и какие методы следует определить как абстрактные.*