

# Задание 1 - Point

Создать класс **Point**

Хранит статик константу

**ZERO (x=0, y=0)** типа Point

Хранит **не** статик

**double x, y;**

Имеет не статик функции:

- **public double calcLength(Point otherPoint)**  
Она возвращает расстояние между двумя точками
- **public double calcLength()**  
Возвращает длину от точки **ZERO** до данной точки  
Необходимо(!) использовать функцию **calcLength(Point otherPoint)** в расчетах
- **public Point normalize()**  
Нормализует вектор(точку). Исходный объект не изменяется. Результат записывается в новый объект.  
Google в помощь о нормализации.  
Необходимо(!) использовать функцию **calcLength()** в расчетах

# Задание 2 - Line

Создать класс **Line**

Хранит не статик

**Point p1, p2;** точки начала и конца линии

**Line lineFx;** из пакета javafx для рисования

Имеет не статик функции:

- **public double calcLength()**  
возвращает длину линии.  
Необходимо(!) использовать -  
- функцию **calcLength(Point otherPoint)** из класса Point
- **public Point getPointOnLine(float percent)**  
**percent** от 0.0f (0%) до 1.0f (100%)  
Возвращает точку, которая находится на данной линии.  
  
0% точка находится в начале линии, то есть на p1  
50% точка находится в центре линии

100% точка находится в конце линии, то есть на p2  
значение **percent** может быть **любым** в диапазоне от 0.0f до 1.0f

Необходимо(!) использовать:

- функцию **normalize()** из класса Point
- функцию **calcLength()** из класса Line

- **public void draw(Pane root)**  
рисует линию
- **public void clear(Pane root)**  
стирает себя (линию)

## Задание 3 - Triangle

Создать класс **Triangle**

Хранит не статик:

**Line l1, l2, l3;** три линии треугольника

Конструктор принимает 3 объекта Point

Имеет не статик функции:

- **public void draw(Pane root)**  
отрисовывает весь треугольник
- **public void clear(Pane root)**  
стирает себя (треугольник)  
Необходимо(!) использовать:
  - функцию **clear()** из класса Line

## Задание 4 - Fractal (дополнительное)

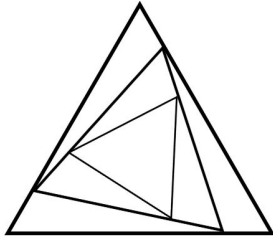
Создать класс **Fractal**

Хранит не статик:

**int depth;** глубина фрактала (кол-во итераций отрисовки)

**float percentOffset;** отступ от начала линии предыдущей итерации отрисовки

Строит фрактал из треугольников.



тут percentOffset примерно равен 0.15f. Глубина = 3

Имеет не статик функции:

- **private void draw(int currentDepth, Triangle triangle)**  
отрисовывает одну итерацию фрактала и вызывает себя рекурсивно до тех пор пока не нарисует все итерации
- **public void draw()**  
отрисовывает весь фрактал
- **public void clear()**  
очищает с экрана данный фрактал  
Необходимо(!) использовать:
  - функцию **clear()** из класса Triangle

## Задание 5 (дополнительное)

Создать класс **FractalInterface**

Хранит в себе не статик **Fractal fractal;**

Этот класс будет уметь отрисовывать окно интерфейса для ввода данных:

- Глубина фрактала
- Процентный отступ по линии (percentOffset)

Кнопки:

- Отрисовать
- Очистить