# Основы разработки приложений на Java

## Уральский производственный центр АйТи – УГАТУ

### Лабораторная работа №1

**Этот вариант можно брать только если это ваши первые лабораторные с ООП :)**

Цели: 1. Освоение синтаксиса и основ ООП в Java.

2. Изучение возможностей стандартных структур данных (коллекций)

Задание.

**1.** Разработать демонстрационную систему классов для работы с геометрическими объектами и векторными рисунками. Классы:   
  
**Точка**: 2 координаты (double!) , метод «Расстояние до другой точки»  
 Необходима функция сравнения точек (переопределить equals класса Object для работы стандартных коллекций)

*Для всех классов определить методы toString() для отладочного вывода.*

**Фигуры** имеют функции вычисления периметра и площади

цвет (строку, html-цвет) и толщину линий

Использовать абстрактный класс.

* **Окружность** (точка + радиус)
* **Замкнутый многоугольник** (список точек, изменяемый)
* **Отрезок**
* **Полилиния** (незамкнутый многоугольник)

Для отрезка и полилинии вычисление площади должно выбрасывать

специальное проверяемое исключение (написать свой класс GeometricException).

У многоугольников и полилиний сделать доступными методы добавления/удаления точек, при этом проверять, нет ли совпадающих точек.

При cовпадении двух точек выбрасывать исключение GeometricException.

1а Реализовать интерфейс или абстрактный класс добавления-удаления точек, общий для многоугольника и полилинии

1b Можно использовать наследование, т.к. методы добавления-удаления общие

**Набор фигур (регион)** – содержит множество фигур, доступных по именам (**Map**).

Можно добавить/удалить фигуру с именем.

Добавить класс-генератор случайных фигур (абстрактная фабрика – метод класса возвращает Figure, а на самом деле – отрезок, треугольник, и т. д.)

Формула для вычисления площади произволного многоугольника:

**2.** Провести тестирование полученной системы (создать наборы тестов JUnit)   
Обязательно протестировать: создание фигур, изменение многоугольника, периметр, площадь, выбрасывание исключений [**@Test(expected=SomeException.class ]**