

Name: _____

Tutor: _____

Matrikelnummer: _____

Punkte: _____

Gruppe: _____

Abgabe: Di, 3.5.2005

2D-Suchbaum

Bei einer Firma die ein CAD-Programm entwickelt, haben sich einige Kunden beschwert, dass das CAD-Programm bei komplexeren Zeichnungen zu langsam arbeitet. Eine Analyse des Programms hat Mängel bei der Verwaltung von Punkten (x, y) in der Zeichnung ergeben.

Ihre Aufgabe ist es nun die verwendete Datenstruktur für einige kritische Operationen zu optimieren. Eine Möglichkeit wäre es einen zweidimensionalen Suchbaum zu verwenden.

```
public class Tree2D extends PointList2D {
    public Tree2D() { ... }

    // Einfügen/Löschen/Finden eines Punktes
    public void insert(Point2D.Double point) { ... }
    public boolean remove(Point2D.Double point) { ... }
    public boolean contains(Point2D.Double point) { ... }

    // Listet alle Punkte auf, die sich im angegebenen Bereich befinden
    public ArrayList findInRangeX(double xFrom, double xTo) { ... }
    public ArrayList findInRangeY(double yFrom, double yTo) { ... }

    // Listet alle Punkte auf, die sich in diesem Rechteck befinden
    public ArrayList findInRectangle(Rectangle2D.Double rect) { ... }

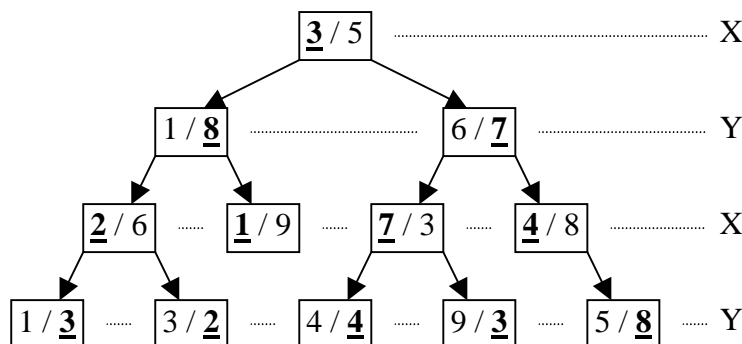
    // Sucht den am nächsten stehenden Punkt (sqrt(dx2+dy2) = Minimum)
    public Point2D.Double findNearest(Point2D.Double point) { ... }

    // Ausgabe aller Punkte (zum Testen)
    public void printAll() { ... }
}
```

Die Klassen `Point2D.Double` und `Rectangle2D.Double` finden Sie im Package `java.awt.geom`, die Klasse `ArrayList` in `java.util`.

Ein 2D-Suchbaum ist im Prinzip ein ganz normaler binärer Suchbaum, nur dass abhängig von der Tiefe des jeweiligen Knotens einmal in x-Richtung und einmal in y-Richtung sortiert wird.

Beispiel:



Mit der Klasse `SimplePointList2D` können anschließend Performancevergleiche durchgeführt werden. Vergessen Sie nicht Ihre Ergebnisse der Performancevergleiche mit abzugeben.