

2. Übung:

Softwarepraktikum C++

Sommersemester 2009

28. April 2009

Abgabe per Email bis Mittwoch, 6. Mai 2009

Aufgabe 2.1:

(Punkte 10)

Segmente:

Implementieren Sie eine Klasse `segment` und statten Sie diese mit einem Intersectionstest aus.

- Implementieren Sie in der Segmentklasse die Funktion `bool intersection(segment& s)`, die ueberprueft ob zwei Segmente einen Schnittpunkt besitzen.
- Benutzen Sie hierzu den Orientationstest, der in der Klasse `point` implementiert ist (Sie benoetigen 4 Orientation Tests).
- Implementieren Sie die Funktion `bool intersection(segment& s, point& inter)`, die den Schnittpunkt `inter` liefert, falls er existiert.

Hinweise:

- Eine Beschreibung des Orientation Praedikats finden Sie in Kapitel 9.2.1 des Leda Buchs.
- Benutzen Sie die Dateien auf der Homepage.

Aufgabe 2.2:

(Punkte 10)

Listen:

Implementieren Sie die folgende Funktionen und geben Sie die asymptotischen Laufzeiten an:

- `int del(int_item*)`
- `int_item* get_item(int i)`
- `void move_to_front(int_item*)`

- `int_item* search(int x)`
- `void conc(int_list& L)`
- `void split(int_item*, int_list& L1, int_list& L2)`

Hinweise:

- Benutzen Sie die Dateien auf der Homepage.
- Die Spezifikationen sind unter der URL http://www.algorithmic-solutions.info/leda_manual/list.html zu finden.
- Ein Tutorial zu LEDA Listen: <http://www.leda-tutorial.org/en/unofficial/ch02s03.html>