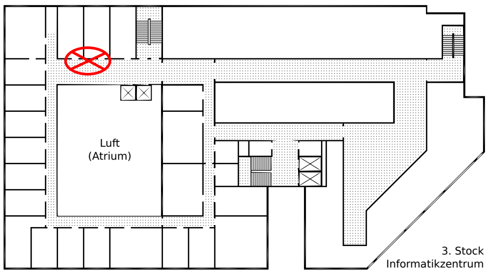
Algorithms Group **Summer '15**
Departement of Computer Science - IBR
TU Braunschweig

Michael Hemmer

Geometric Algorithms
Exercise 2
May 13, 2015



This sheet is comprised of several exercise. The solutions should be handed in by **May 21, 2015** until 19:00pm. This can be done by placing them in the appropriate box of the exercise locker, see floor plan on the right.
In order to achieve the *"Studienleistung"*, you must achieve 50% of the points over all exercise sheets and you must have presented at least two exercises until the end of the term. Please mark those exercises that you would like to present during the tutorial.

Exercise T1 (Do Intersect):

The intersection detection problem for a set S of n line segments is to determine whether there exists a pair of segments in S that intersect, i.e., the algorithm simply returns `true` or `false`. Give an algorithm that solves the intersection detection problem in $O(n \log n)$ time.

Is this algorithm output sensitive ?

(8+2 P.)

Exercise T2 (Output Sensitive – Sweep Line):

Show that Algorithm 1.15 `FindIntersections` is also output sensitive if it only outputs the intersection points, i.e., without reporting the involved line segments at each intersection point. That is, you must show that

$$O((n + k) \log(n)) = O((n + I) \log(n)),$$

where n is the number of input segments, I the number of intersection points and k the output size of the original Algorithm 1.15.

(10 P.)

Exercise T3 (O(n) Space – Sweep Line):

Change Algorithm 1.15 `FindIntersections` such that the working storage is $O(n)$ instead of $O(n + k)$, where k is the number of intersections. The working storage is the space that is actually required by the algorithm, i.e, already reported intersections that the algorithm does not explicitly keep in a data structure are not counted.

Prove that the new algorithm requires $O(n)$ space and argue why the time complexity is not effected.

Can you come up with an input in which the original Algorithm 1.15 requires more than $O(n)$ space or can you proof that it always requires only $O(n)$ space?

(10+5+5* P.)

Exercise T4 (Batched Point Location - Sweep Line):

Given the DCEL of a planar subdivision S and a set of points P. Give an algorithm that locates every point of P within S, i.e., for every point $p \in P$ report p and the feature that p is located in. The algorithm should run in $O((n+m)\log(n+m))$ time, where n is the complexity of S and m is the cardinality of P.

Give also a running time analysis of your algorithm.

(10+5 P.)