Welcome to 02110. All practical information about the course can be found on the course webpage.

Structure

The course runs on Thursdays 8-12. Exercise class is from 8.00-10.00 and lectures from 10.00-12. There is a weekplan for each week containing information about literature for the week and exercises. The weekplans are placed under the week, where the lecture in the subject is given. Then you work with the exercises at the exercise class the week after.

The class is structured as follows:

- **8.00-9.15 Group work.** Time to work on the exercises you couldn't solve at home. The TAs will be there to help you.
- **approx. 9.25-9.45 Walk-through of solutions to the exercises.** Together with the TA you will go through the solutions to the exercises in class. You are expected to have already solved the exercises and you should be prepared to discuss your solutions with the rest of the class.
- **10.00-11.15 Lecture**
- **approx. 11.15-11.45 Work on exercises in the material from the lecture**
- **11.45-12.00 Round up**

Textbook

KT: "Algorithm Design" by Kleinberg ad Tardos. Additional literature will be supplied on CampusNet.

Exam

The exam is a written exam. You are not allowed to bring your computer to the exam.

Mandatory assignments

In order to be allowed to participate in the written exam it is a requirement that you:

- get at least 50 points in the mandatory written assignments.
- get at least 50 points in the mandatory programming assignments on CodeJudge.

Each week from week 2 to 11 you will be asked to do written mandatory exercises and hand them in. These exercises will be corrected by the teaching assistants. Each weekly exercise can give up 20 points. 20 points is given for a perfect solution, and you can get anything between 0 and 20 points for a hand-in depending on the quality of your solution and your writing. The exercises do not count in the final grade for the course, but you have get at least 50 points in order to be allowed to participate in the exam. Start early, do not wait until the last 2 or 3 weeks before you hand in exercises. The written assignments can be found on DTU Learn.

Approximately every second week there is an implementation assignment that should be handed in in CodeJudge. You can get up to 20 points. The deadlines for the mandatory assignment (both written and programming assignments) is always Sunday at 20:00. The deadline for handing in the assignments must be respected.

Collaboration policy for mandatory assignments

Remember to read the collaboration policy on the webpage.

How do I prepare for class?

You are expected to have looked at all exercises at home and to have solved most of them, and you should be prepared to present your solutions to the rest of the class.

We also assume that you read the material for the week's lecture. You can read it before or after the lecture, that's up to you. The lecture is meant to clarify and give an overview — it will not necessarily cover all material for the week in details.

It is a good idea to hand in as many homework assignments as possible. The questions will be similar to the exam questions, so it is a good opportunity to get some feedback on you writing.
**Lecture and literature**

For reading material etc for the first lecture see weekplan 1.

**Exercises**

The exercises on this week plan is for the exercise class in Week 1 at 8:00, i.e., before the first lecture. English speaking students should go to room 358/042.

1 **Union and Intersection**  Suppose you are given two sorted lists $A$ and $B$ of integers of length $n_A$ and $n_B$, respectively.

1.1 Describe an algorithm that computes the set $C = A \cap B$. The output should be a sorted list $C$ containing all elements that are in both $A$ and $B$.

1.2 Describe an algorithm that computes the set $C = A \cup B$. The output should be a sorted list $C$ containing all elements that are in either $A$ and $B$.

2 **Pancake sorting**  Suppose you are given a stack of $n$ pancakes of different sizes. You must sort the pancakes so that smaller pancakes are on top of larger pancakes. The only operation you can perform is a flip: insert a spatula under the top $k$ pancakes, for some integer $k$ between 1 and $n$, and flip them all over.

2.1 Describe an algorithm to sort an arbitrary stack of $n$ pancakes using $O(n)$ flips. Exactly how many flips does your algorithm perform in the worst case?

2.2 For every positive integer $n$, describe a stack of $n$ pancakes where your algorithm from 2.1 requires $\Omega(n)$ flips to sort.

3 **Subsequence**  A string $P$ is a subsequence of string $T$ if we can obtain $P$ from $T$ by removing 0 or more characters in $T$. For instance, $aba$ is a subsequence of $bcadfbbba$.

3.1 Give a linear time algorithm that decides whether a string $P$ is a subsequence of a string $T$.

3.2 Given $k$ strings $P_1, P_2, \ldots, P_k$, and a string $T$, give an algorithm that decides which of the $k$ strings that are subsequences of $T$.

4 **Interval Scheduling**  In the interval scheduling problem you are given a set of requests labeled $1, \ldots, n$. Each request has a starting time $s_i$ and a finish time $t_i$. A subset of requests are compatible if no two of them overlap in time. The goal is to select a subset $S$ of compatible intervals that is as large as possible. Give an algorithm that solves the interval scheduling problem.

5 **Networks**  Consider a diagram of a telephone network which consists of switching centres and communication lines. Each communication goes between a pair of switching points, and has an associated bandwidth and a cost.

5.1 Give an algorithm that given two switching centres $a$ and $b$, outputs the cost of a minimum cost path between $a$ and $b$.

5.2 The bandwidth of a path of communication lines between two centres is the lowest bandwidth on the path. Give an algorithm that given two switching centres $a$ and $b$ outputs the maximum bandwidth of a path between $a$ and $b$.  

---

2
Puzzle of the week: 99 Cops  A town has 99 cops. A cop is either honest or corrupt, the majority of the cops is honest. You need to figure out all the corrupt cops, with less than 299 questions. All cops know who is honest and who is corrupt, but only honest cops will answer truthfully. Corrupt cops may lie arbitrarily. For security reasons you can only ask one type of question: You may ask cop X whether cop Y is corrupt. This question will by answered by X with either "Y is corrupt" or "Y is honest".