



---

## INF2220: algorithms and data structures

---

### Series 12

Topic Kombinatorisk søk, beregnbarhet og kompleksitet

Issued: 10. 11. 2017

### Classroom

**Exercise 1 (TV cables)** Et firma som leverer TV-kabler til boligfelt har to typer kabler, A og B. Kabelen av type A er litt billigere, men til gjengjeld kan den ikke kuttes opp, men må legges i EN sammenhengende sløyfe som er innom alle husene, og som til slutt går tilbake til det punktet der den startet.

Kabelen av type B kan derimot deles opp og legges i stykker mellom husene. Kabelstykkene må legges fra hus til hus slik at alle husene er forbundet med hverandre (direkte eller indirekte).

For å kunne gi gode tilbud vil firmaet bruke så lite kabel som mulig, og også beregne om det lønner seg å bruke kabeltype A eller B. Som en forberedelse til en leveranse vil de alltid dra ut på det aktuelle hus-feltet og måle opp for hvert par av hus hvor mye kabel det vil gå med for å strekke kabel direkte fra det ene huset til det andre. Vi lar  $n$  angi antall hus på feltet.

Ola og Kari jobber i firmaet. Ola påstår at han har en algoritme med tidsforbruk  $O(n^3)$  som kan finne hvordan man kan legge kabel av type A slik at man bruker kortest mulig kabel. Kari påstår tilsvarende at hun har en algoritme med tidsforbruk  $O(n^2)$  som finner hvordan man kan legge kabel B for å bruke minst mulig av denne.

Vurder påstandene til Ola og Kari. Begrunn svaret.

**Exercise 2 (Baseball cards)** Exercise 9.56 in MAW.

### Lab

**Exercise 3 (Selection)** Write a program that prints out all possible selections with exactly  $m$  elements from the set of numbers  $0, 1, 2, \dots, n - 1$  (assuming  $m \leq n$ ).

**Exercise 4 (Derangements)** A *derangement* is a permutation  $p$  of  $\{1, \dots, n\}$  such that no item is in its proper position, i.e.  $p_i \neq i$  for all  $1 \leq i \leq n$ . Write an efficient backtracking program with pruning that constructs all the derangements of  $n$  items.

**Exercise 5 (Bridge crossing)** There are four people (A, B, C, D) have to cross a bridge. However, the bridge is fragile and can hold at most two of them at the same time. Moreover, to cross the bridge a torch is needed to avoid traps and broken parts. The problem is that these people have only one torch that lasts for only 60 minutes. Each one of them needs a different time to cross the bridge (in either direction):

- A 5 minutes
- B 10 minutes
- C 20 minutes
- D 25 minutes

The problem is now: In which order can the four people cross the bridge in time (that is, in 60 minutes)?

**Exercise 6 (Number placement)** (From Krogdahl and Maus). Find permutations  $x_1, x_2, \dots, x_9$  of the nine decimal numerals 1, 2, ..., 9 which satisfies the following condition: the decimal number  $x_1x_2$  is divisible by 2, the number  $x_1x_2x_3$  is divisible by 3, ..., analogously up to  $x_1x_2 \dots x_9$ . Find all such permutations.

**Exercise 7 (Land colouring)** We want to colour the countries on a map, such that countries with a common border do not have the same colour. This can always be done with four colours. Given  $n$  countries (numbered from 0 to  $n - 1$ ) and a boolean  $n \times n$ -array indicating for all pairs of countries whether they have a common border or not.

The task is to make a program that creates a possible colouring using only four colours (numbered from 0 to 3).

**Exercise 8 (N-Queens)** In the well-known *eight queens problem*, the challenge is to place eight queens on an 8 x 8 chessboard so that no queen can take another one, i.e., no *two* queens share the same *row*, *column*, or *diagonal*. The eight queens problem is an example of the more general *N-queens problem* of placing  $n$  queens on an  $N \times N$  chessboard. Write an implementation to solve the *N-queens problem*, where  $N \in \mathbb{N}$  and  $N > 3$ .