



3. *Vi ses i morse!*

I en fil med morsemeddelanden vill du söka efter ordet **ses** som i morsekod blir  $\dots\blacksquare\cdot\blacksquare\cdot\cdot\cdot\blacksquare\blacksquare$ , där  $\cdot$  är ett kort pip och  $\blacksquare$  står för tystnad.

(5p)

Rita en Knuthautomat för denna elva tecken långa söksträng och skriv upp backvektorn. Vad tror du om snabbheten om du jämför med Boyer–Moore's algoritm?

4. *Sudokulösaren*

(8p)

Det är äntligen dags att tänka ut hur din sudokulösare ska programmeras. Funktionen ska vara följande: Man ska kunna läsa in en sudoku som nio nioteckensrader (siffror och blanka, inga nollor) och sedan ska programmet skriva ut alla lösningar. Beskriv lämpliga algoritmer och datastrukturer!

5. *Nivåutskrift*

(4p) Ett binärträd kan ju skrivas ut från vänster till höger i inorder. Nu vill vi ha samma utskrift av en viss nivå i trädet. Anropet skulle kunna se ut som `tree.write(17)` om man ville se sjuttonde nivån (roten är nivå noll). Ge en rekursiv tanke för detta!

(2p) Samma uppgift för ett allmänt träd representerat med `down`- och `right`-pekare!

6. *Komprimerat personnummer*

(4p) Olika siffrors förekomst i textfiler anges i tabellen. Rita motsvarande huffmanträd och binärkoda ditt personnummer efter trädet.

0	32%
1	20%
2	10%
3	12%
4	7%
5	2%
6	4%
7	4%
8	3%
9	6%

(2p) Hur många (åttabitars) byte får man om man komprimerar en sifferfil på en kilobyte (tusen siffror)?

7. *Gröna granna grangrenar*

- (5p) En perfekt formad gran kännetecknas av att grenarna blir kortare ju högre upp dom sitter. Varje gran kan göras perfekt om man skär bort några grenar. Uppfinn en algoritm som givet grenlängderna uppifrån och ner bestämmer hur gör man den perfekt genom att knipsa så få grenar som möjligt!

8. *Syntax för citatcitat*

” UR ’ UR ” DET KOMMER I ’ I ” TIDIGARE ’ HÄFTEN’ ” ’ ” ’ ”

(6p) Detta namn på en skrift citeras i Povel Ramels ”Min galna hage”. Som synes växlar man mellan enkla och dubbla citattecken när man har flera citeringsnivåer.

Skriv en syntax för text med denna konvention. Du kan använda symbolen `<text>` för att beteckna godtycklig text utan citattecken.

9. *DNA-spår*

(4p) När polisen hittar DNA-spår på brottsplatsen vill den gärna kolla om det finns i det stora DNA-register som ska byggas upp. För att få göra en sökning krävs beslut av domstol, som tar hänsyn till brottets svårighetsgrad med mera.

För det mesta ger sökningen ingen träff, och då var den tidskrävande domstols-proceduren meningslös. Polisen skulle vilja ha ett förhandsbesked om huruvida DNA-spåret finns i databasen. Uppfinn en metod som ger denna möjlighet utan databassökning och utan att någon DNA-profil kopierats över till någon ny databas.

En DNA-profil kan vara ett 64-bitars ord, där varje etta betyder att en viss delsekvens finns med.

(2p) Försök också att komma på en metod som kollar om det finns nästanträffar, alltså avvikelser på högst en bit.