



Fredag 20 oktober 2017 kl 8–12

Hjälpmedel: En tryckt algoritm-/datastruktursbok och ditt eget formelblad. För betyg E krävs att alla E-uppgifter är godkända (upp till två E-uppgifter kan kompletteras). För betyg D krävs (utöver E-kraven) D på bägge C-uppgifterna, för C krävs minst ett C och ett D (en C-uppgift kan kompletteras till D). För betyg B respektive A krävs (utöver C-kraven) betyg B respektive A på A-uppgiften (A-uppgiften kan inte kompletteras). Lycka till!

1. *Fest på fars dag*

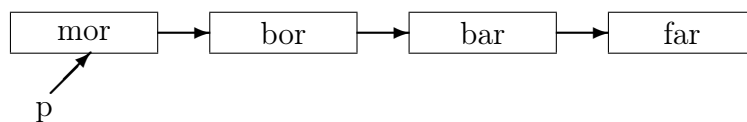
Betyg E. Om ett par veckor är det fars dag. Konstruera en KMP-automat som söker efter PAPPAPARTY Rita upp automaten och ange next-vektorn.

(15 min)

2. *Parent-pekare*

Betyg E. I fjärde labben löste du problemet att hitta kortaste vägen från ett ord till ett annat, t ex från **far** till **mor**. I femte labben använde du sedan noder med parent-pekare för att hålla reda på de ord som gav lösningen. Med en rekursiv utskriftsfunktion kunde du skriva ut orden i rätt ordning (t ex **far bar bor mor**). Hur gick den *rekursiva utskriften* till? Rita och förklara!

```
class ParentNode:
    def __init__(self, word, parent = None):
        self.word = word
        self.parent = parent
```



(10 min)

3. *Kryptering*

Betyg E. Här är två krypteringsmetoder vi tagit upp i kursen:

- One-time pad
- RSA (Rivest-Shamir-Adleman)

Är det viktigt med säker nyckelöverföring för bägge dessa metoder? Förklara!

(10 min)

4. *Hasha pappor*

Betyg E. BVC ordnar pappagrupper för nyblivna fäder. Papporna ska hashas in i en hash-tabell på följande vis:

- Hashtabellen har 13 platser
- Vid krock används krocklistor
- Hashfunktionen är summan av siffrorna i pappans födelsedag och månad. Den som är född 23/10 får alltså hashvärdet $2 + 3 + 1 + 0 = 6$.

Rita upp hashtabellen, med eventuella krocklistor, för följande åtta pappor:

- Amir 28/10
- Bedrich 1/6
- Calvin 15/5
- Dubaku 11/11
- Ejvind 21/12
- Filip 23/2
- Gus 2/5
- Henrik 19/1

(15 min)

5. *Komplexitet*

Betyg E. Följande sex tidsfunktioner är givna:

- a) $A(n) = 7904 \cdot n \cdot \log_2(n)$
- b) $B(n) = 876 \cdot n^2 + 28 \cdot n + 3 \cdot n \cdot \log_2(n)$
- c) $C(n) = 7904 \cdot n + 17019$
- d) $D(n) = 5 \cdot n^2$
- e) $E(n) = 245 \cdot \ln(n)$
- f) $F(n) = 714 \cdot 2^n$

- a) Berätta vilken Ordo-klass var och en av funktionerna ovan tillhör.
- b) Rangordna sedan Ordo-klasserna - den som växer långsammast först, sen den näst långsammaste osv.

(15 min)

6. *Syntax för faderskapstest*

Betyg E. Här är en syntax för en rad i ett faderskapstest:

```
<Rad> ::= <Locus> " " <Mamma> " " <Barn> " " <Pappa>
<Mamma> ::= <Par>
<Barn> ::= <Par>
<Pappa> ::= <Par>
<Par> ::= <Antal> "," <Antal>
<Locus> ::= "D5S818" | "vWA" | "CSF1PO" | "TH01"
<Antal> ::= [1-9] [0-9]?
```

Vilka rader godkänns av syntaxen, och vilka godkänns inte? Motivera dina svar!

- a) TH01 06,07
- b) D5S818 vWA CSF1PO TH01
- c) CSF1PO 19,21 19,19 10,21
- d) vWA 15,16 CSF1PO 14,14 TH01 7,9

(15 min)

7. *Komprimering*

Betyg C.

Ἀλέξανδρος; Alexander; اسلامكبر; அலெக்சანდரே; アレクサンダー; אלכסנדר; Ἀλέξανδρος; அலக்ஷத்ரி; सिकंदर; 알렉산더; அலெக்சாண்டர்

Anta att vi vill lagra en lista med namnen på all världens pappor. Namnen ska skrivas med rätt tecken ur respektive skriftspråk (du kan anta att det finns omkring 100 000 olika tecken totalt i skriftspråken).

Listan blir lång, så vi vill komprimera den, kanske med någon av metoderna Huffmankodning eller Lempel-Ziv (LZW).

- a) Skriv upp två egenskaper som är intressanta att jämföra för komprimeringen.
- b) Jämför Huffmankodning och Lempel-Ziv för var och en av de två egenskaperna. Motivera dina slutsatser och illustrera med exempel. Du får göra egna antaganden om namnlistans innehåll.

För full poäng krävs en välstrukturerad, välmotiverad och begriplig jämförelse.

(30 min)

8. *Heap eller binärträd?*

Betyg C. Många blandar ihop datastrukturerna heap och binärt sökträd. Nu ska du hjälpa till att förklara skillnaden!

- a) Visa hur en heap är uppbyggd. Rita upp ett exempel.
- b) Visa hur ett binärt sökträd är uppbyggt. Rita upp ett exempel.
- c) Hur gör du för att avgöra om en struktur är en heap eller ett binärt sökträd? Berätta om två olika sätt.
- d) Beskriv ett problem där det passar bäst att använda binärt sökträd. Förklara varför.
- e) Beskriv ett annat problem där det passar bäst att använda heap. Förklara varför.

(30 min)

9. *Lukes pappas problem*

Betyg A.

Luke Skywalkers pappa Darth vill ta sig till rebellernas gömställe så snabbt som möjligt. Hans rymdskepp rör sig genom att hoppa genom de maskhål som förbinder vissa punkter i rymden.

Om antalet punkter är n så kan kopplingarna beskrivas av en boolesk $n \times n$ -matris. $connected[p_1][p_2]$ är True om det finns ett maskhål mellan punkterna p_1 och p_2 .

- a) Beskriv en *effektiv* algoritm som hittar och skriver ut den kortaste vägen (så få hopp som möjligt). Algoritmen måste beskrivas på ett strukturerat och tydligt sätt.
- b) Berätta vilka datastrukturer du använder.
- c) Demonstrera hur din algoritm fungerar med ett litet exempel.
- d) Uppskatta tidskomplexiteten för din algoritm.

(45 min)