

Fredagen den 24 oktober 2014 kl 9–13

Hjälpmedel: En algoritmbok och ditt eget formelblad. För betyg E krävs att alla E-uppgifter är godkända, för betyg C att alla E- och C-uppgifter är godkända, för betyg A att alla uppgifter är godkända.

1. *Sök cykelram*

Betyg E. Konstruera och rita upp en KMP-automat som söker efter ordet RACERRAM. Ange även next-vektorn!
(10 min)

2. *Välj cykelnyckel*

Betyg E. Tildas cykelbutik har ett stort antal cyklar till försäljning. Välj någon egenskap hos cykeln som nyckel, och visa med ett eget exempel (rita och illustrera) hur man kan sortera cyklarna med mergesort.
(15 min)

3. *Tänd cykellyse*

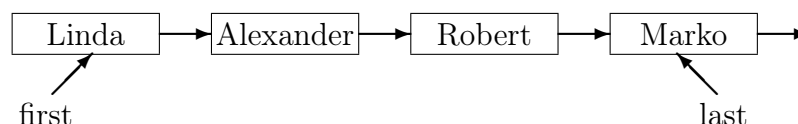
Betyg E. Nu när det blir mörkare om kvällarna känns det extra viktigt att ha lysen till cykeln. Du har lagt in lamporna med priset som prioritet i en *min*-heap. På vektorform ser heapen ut så här:

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 10 | 40 | 30 | 42 | 41 | 48 | 50 | 49 |
|----|----|----|----|----|----|----|----|

Rita upp heapen på trädform och visa sen hur det ser ut när man plockar ut två element (du vill ju inte köpa de allra billigaste). Visa minst fem steg. Skriv slutligen upp heapen på vektorform igen.
(15 min)

4. *Bilda kö*

Betyg E. Cyklisterna köar vid övergångsstället nere vid Valhallavägen.



Kön är implementerad med en länkad lista. Rita bilder som visar hur det ser ut i detalj (varje steg) när man sätter in respektive plockar ut ett element ur kön.
(10 min)

5. Fyll cyklistträdet.

Betyg E. Vad är tidskomplexiteten för att stoppa in n cyklister i ett binärt sökträdet? Du kan förutsätta att trädet (av en lycklig slump) alltid kommer att vara balanserat.
Motivera dina beräkningar.
(10 min)

6. Vanligaste bokstäverna

Betyg E.

| | | | | | | |
|----|----|----|----|----|---|---|
| E | A | N | R | T | S | I |
| 25 | 25 | 13 | 12 | 12 | 7 | 6 |

Rita upp ett Huffmanträdet för tabellen ovan, och skriv upp vilka koder bokstäverna får.
(10 min)

7. Koda kort

Betyg C. Följande tre uppsättningar av koder är tänkta att använda för komprimering. Vilka fungerar för komprimering? Vilken är effektivast? Motivera ditt svar!

(20 min)

| | kod1 | kod2 | kod3 |
|---|-------------|-------------|-------------|
| E | 11 | 111 | 10 |
| A | 10 | 101 | 11 |
| N | 011 | 100 | 110 |
| R | 010 | 011 | 101 |
| T | 001 | 010 | 111 |
| S | 0001 | 001 | 0100 |
| I | 0000 | 000 | 0111 |

8. Cykel eller inte cykel?

Betyg C. Tilda har en lång lista med termodynamiska processer och deras egenskaper. Hon vill snabbt kunna slå upp om en process är cyklisk, och hon planerar därför att skapa en hashtabell.

Hennes förslag är att skapa en boolesk hashtabell, där man hashar in processerna ungefär så här:

```
for process in lista:
    index = hashfunktion(process.namn)
    if process.cyklisk == True:
        tabell[index] = True
    else:
        tabell[index] = False
```

Jämför denna idé med att använda en vanlig hashtabell.

(15 min)

9. Krypterat cykelskvallet

- Betyg C.** Sedan 1896 har Tilda och Osquar diskuterat cyklar via bokchiffer - dom äger bägge en originalutgåva av Archibald Sharps *Bicycles & Tricycles: A Classic Treatise on Their Design and Construction*. Nu funderar Tilda på att uppgradera till RSA. Vilka egenskaper är viktiga hos en krypteringsmetod? Räkna upp dessa och hjälp Tilda att göra en jämförelse mellan bokchiffer och RSA.
(20 min)

10. Bygg cykelboklista

- Betyg A.** Tilda vill bygga en lista HOJ96 med all cykellitteratur skriven före 1896. Hon kommer på idén att utgå från sin favoritbok (se ovan) och använda referenser för att bygga listan. Archibalds bok inleder förstas listan, sedan följer (i valfri ordning) innehållet i dess referenslista, sedan referensernas referenslistor osv. Formulera en rekursiv tanke för listbygget av HOJ96. Du behöver inte ta hänsyn till att samma referens kan förekomma flera gånger.
(25 min)

11. Cykla ledsmart

- Betyg A.** Cykelsällskapet har en databas med sina 24 cykelleder genom Sverige. Varje led är en följd av ortnamn med kilometerangivelser för varje etapp, t ex ... Gemla 18.9, Växjö 10.5, Nöbbele 15.2 ... Lederna korsar varann i vissa orter och den kortaste vägen går ofta på flera leder. Beskriv en algoritm som finner kortaste vägen mellan två orter. Algoritm-beskrivningen måste vara korrekt, välstrukturerad och begriplig. Glöm inte att ange vad som är indata och utdata till algoritmen. Förklara till sist varför algoritmen fungerar och ange vilka datastrukturer som behövs vid implementeringen.
(30 min)

