

2D1320, TENTAMEN I TILLÄMPAD DATALOGI

Lördagen den 19 oktober 2002 kl 14–19

Maxpoäng = 50. Betygsgränser: 25 poäng ger trea, 35 ger fyra, 45 ger femma.

Resultatet anslås senast 1 november på Nadas anslagstavla.

Hjälpmedel : En algoritmbok och det rosa formelbladet

1. *Spöksjö*

- (5p) Konstruera och rita upp en KMP-automat som söker efter ordet **VÅLNADSVÅLD**.
Glöm inte att ange next-vektorn!

2. *Binärt sökträd*

- (2p) Rita upp det binära sökträd som bildas om man i tur och ordning stoppar in följande ord i trädet:

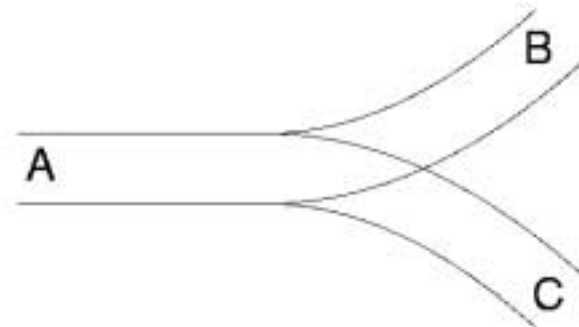
ande fantom gast gengångare poltergeist spöke vålnad

- (3p) Hur många jämförelser krävs för att upptäcka att ordet *vampyr* inte finns med i ditt träd? Rita ett balanserat sökträd för de sju orden ovan.

- (5p) Formulera en *rekursiv* tanke för att skriva ut enbart löven i ett binärt träd. Välj själv i vilken ordning dom ska komma ut. Vad kommer att skrivas ut för det balanserade trädet du ritade ovan?

3. *Spöktåg*

Spöktåget har sett likadant ut de senaste hundra åren. För att få lite variation vill man nu ordna om bland vagnarna (tåget har inget lok, så vilken vagn som helst kan vara först). Som tur är finns det två stickspår som man kan köra in vagnar på (dom är nämligen för tunga för att lyfta).



- (6p) Beskriv utförligt en algoritm som flyttar om vagnarna till önskad permutation. Spåren A, B och C representeras av tre *abstrakta stackar*. De n vagnarna står från början på spår A, och önskad permutation finns lagrad i vektorn *önsketåg*[n]. Efteråt ska önsketåget stå på spår A.

4. *Spökenas tio i topp*

- De ambitiösaste tusen spökena har ordnat en tio-i-topp-lista, där de tio spöken som skrämt flest personer på en dag står med. Listan ska uppdateras varje dag. Vilka av följande metoder är användbara och varför/varför inte? Räknesortering (distributionsräkning), quicksort, insättningsortering, urvalssortering.
- (4p)
- (2p) Vilken algoritm är bäst i detta fall? Motivera!

5. *Spökhuset*

- Du vaknar mitt i natten och upptäcker att du förflyttats till ett spökhuis! Huset är enormt och det finns många utgångar, men det gäller att komma ut snabbt, innan spökena hittar dig.
- (8p) Beskriv en algoritm som på ett så effektivt sätt som möjligt tar reda på den kortaste vägen ut ur spökhuset. Du har tillgång till en karta över huset.
- Du behöver inte skriva någon programkod, men du ska förklara algoritmen utförligt och beskriva datastrukturer, metoder och klassuppdelning.

6. *Andemeningen*

Hemsökarnas utbildningsutskott (HUU) vill lära ut det korrekta sättet för väsen att väsnas på. Det finns endast två korrekta andemeningar: vrål, som börjar på B, följt av valfritt antal U, och skratt, som består av en tvåbokstavskombination (H följt av en valfri vokal) upprepade valfritt antal gånger. Exempel på godkända andemeningar är:

```
BU
BUUUU
HÖHÖHÖ
HAHAHIHI
```

- (6p) Skriv en syntax för en andemening.

7. *Spökhistoria*

- Man vill lagra spökhistorier från 1800-talets mitt och framåt i en databas, där man ska kunna söka på ort eller spöktyp, t ex Jämtland eller Mylingar. Hur ska hashningen gå till och hur stor behöver hashtabellen vara?
- (4p)

8. *Spökposter*

- Du tänker skriva ett litet äventyrsspel och funderar på hur du bäst representerar ett spöke. Ett heltal som anger skrämseffekt kanske?
- (3p) Motivera varför det är bäst med en abstrakt datatyp för ett spöke. Ge också exempel på metoder med deras namn, in- och utdatatyper, t ex
- (2p) `public utdatatyp metod(indatatyp variabelnamn)`