

## 2D1320, TENTAMEN I TILLÄMPAD DATALOGI

Lördagen den 21 mars 1998 kl 9–14

Maxpoäng tenta+bonus = 50+5. Betygsgränser: 25 poäng ger trea, 35 ger fyra, 45 ger femma.

Resultatet anslås senast 28 mars på Nadas anslagstavla.

Hjälpmedel : En algoritmbok.

### 1. Knuthautomat

Datalogiprofessor Knut Donald vill veta om hans födelsedatum 330331 finns i mastodontverket *The art of computer programming*, vars textmassa ligger på en fil.

(6p) Hur såg hans sökautomat för 330331 ut? Rita upp den med heldragna framåtpilar och prickade bakåtpilar och ange den `next`-vektor som definierar automaten!

(2p) Knut har bråttom, eftersom han pensioneras om tio dagar! Vilken sökmetod skulle förmodligen ha varit den snabbaste? Ungefär hur många gånger snabbare kan den vara?

### 2. Ordna stacken efter kön

(6p) Personposter har lagts in i en stack, ordnade på personnummer så att äldre personer ligger under yngre personer. Man vill organisera om den så att alla män ligger under alla kvinnor, men fortfarande äldre män under yngre män och äldre kvinnor under yngre kvinnor.

Beskriv utförligt en algoritm för att ordna om stacken på detta sätt. Till din hjälp har du en kö och både stacken och kön är abstrakta.

### 3. Rekursiv spegling

(6p) Ett binärträd med personposter är ordnat med mindre personnummer till vänster och större till höger. Nu vill man bygga om trädet så att det blir precis tvärtom. Efter satsen `Spegla(rot)` ska trädet under pekaren `rot` ha speglats på detta sätt. Beskriv i ord eller i valfritt programspråk en rekursiv tanke för `Spegla`.

### 4. Ordlistig hashning

(3p) För att snabbt kunna avgöra om ett ord finns i SAOL vill man använda en hashvektor med krocklistor (separate chaining). Följande hashfunktioner har föreslagits. Är någon av dem klart bättre än den andra? Motivera! Som exempel ges hashvärdet för `AB`.

a) ASCII-värdenas produkt modulo 158773.  $AB \mapsto 65 \cdot 66.$

b) ASCII-värdena hopsatta till ett tal, modulo 158773.  $AB \mapsto 6566.$

(3p) Viggos stavaprogram använder inte krocklistor utan i stället fjorton olika hashfunktioner. Förklara principen!

## 5. *Problem i patiensbok*

”Till denna uppgift används alla ess, kungar, damer och knektar, alltså sexton kort. Det gäller att lägga ut korten i en  $4 \times 4$ -matris så att två kort av samma färg eller valör aldrig finns i samma rad, kolumn eller någon av de båda diagonalerna.”

Man vill att ett program ska skriva ut alla lösningar på nedanstående sätt.

```
hj E   sp D   kl Kn  ru K
kl K   ru Kn  hj D   sp E
ru D   kl E   sp K   hj Kn
sp Kn  hj K   ru E   kl D
```

- (8p) Du behöver inte skriva programkod, men du ska förklara algoritmen utförligt och beskriva datastrukturer, procedurer och moduluppdelning.

## 6. *Syntax för funktionsanrop*

$F(x, P(a, b), y, G(z, H(a), x, P(y, y)), F(F(F(c))))$

- (6p) Skriv en kontextfri grammatik för funktionsanrop av denna typ. Funktionsnamn är alltid en stor bokstav, variabelnamn alltid en liten bokstav. Alla anrop ska ha minst en parameter men syntaxen kollar som synes inte att varje anrop till en funktion har samma parameterantal. Använd till exempel symbolerna  $\langle \text{funktionsanrop} \rangle$ ,  $\langle \text{variabel} \rangle$ ,  $\langle \text{funktion} \rangle$ ,  $\langle \text{parameterlista} \rangle$ .

Beskriv i ord hur ett program kan undersöka om ett funktionsanrop följer din syntax! Vilka procedurer behövs?

## 7. *Tio i topp*

Femtio gånger per sekund når spänningen i vägguttaget ett maxvärde och via en AD-omvandlare skickas det till en pc. Man vill att programmet varje dygn ska rapportera dom tio högstanoteringarna och när dom inföll. Följande datastrukturer har föreslagits för uppgiften:

- En vektor med 4,32 miljoner poster som insättningssorteras.
- En vektor med 4,32 miljoner poster som quicksorteras vid dygnets slut.
- En hashvektor med 6,48 miljoner pekare och hashning på spänningsvärdet.
- En trappa (heap) med plats för 4,32 miljoner poster.
- En trappa med plats för tio poster.
- Ett binärträd, sorterat efter spänningsvärdet.

- (6p) Vilka av dessa strukturer är odugliga? Hur rankar du dom övriga när det gäller komplexitet? Motivering krävs!

## 8. *Gyllene tider*

- (4p) Många datorprogram som lagrar årtal med bara två siffror kommer att ge problem vid sekelskiftet. Att skriva om eller byta ut programmen kommer att ge västvärldens programmerare hundratals miljarder kronor i arbetsinkomst.

Om programmen använt en abstrakt datumtyp hade inkomsten krympt till en tusendel eller så. Förklara! Ge exempel på vad en abstrakt datumtyp bör innehålla och hur den kan implementeras?