

2D1320, TENTAMEN I TILLÄMPAD DATALOGI
Fredagen den 16 januari 2004 kl 8–13

Maxpoäng = 50. Betygsgränser: 25 poäng ger trea, 35 ger fyra, 45 ger femma. Resultatet anslås senast 13 februari på Nadas anslagstavla. Hjälpmedel : En algoritmbok och det rosa formelbladet.

1. *Televisionens tio-i-topp*

Resultatet av en tittarundersökning med en miljon deltagare ska publiceras i form av en tio-i-topp-lista. Följande datastrukturer och sorteringsalgoritmer har föreslagits för uppgiften:

- En vektor med en miljon element som sorteras med mergesort.
- En stack som quicksorteras.
- En länkad lista som insättningssorteras.
- En trappa (heap) med plats för en miljon element.
- En trappa med plats för tio element.
- Ett binärträd.

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

2. *Teori*

(10p) Nedan finns fem frågor om algoritmer och datastrukturer. Varje fråga kan ge upp till två poäng. Motivering krävs!

- Är rekursion snabbare än iteration?
- Kan ett binärt träd implementeras med en array?
- Passar KMP-automater bra för webbsökning?
- Räcker en hashvektor med N platser för att hasha in N element om man har en perfekt hashfunktion?
- Kan förstörande komprimering (lossy compression) användas för bilder?

3. *Telepatisyntax*

(3p) Kort med symbolerna $\circ+$ \approx $\square*$ används för att testa telepatisk förmåga. Skriv en syntax för en kortföljd med ett eller flera kort.

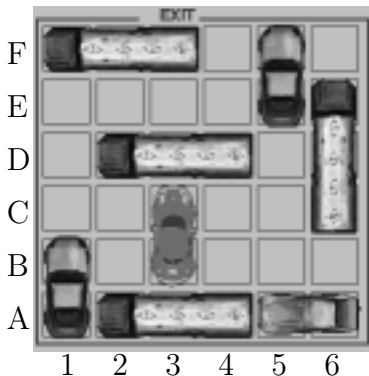
4. *Telebin*

Med den nya binära SMS-tjänsten Telebin, där man betalar per bit, kan man komma billigare undan om man själv komprimerar sina meddelanden.

(6p) Rita upp ett Huffmanträd för meddelandet **KOMPRIMERA MERA** och skriv sedan meddelandet Huffmankodat.

5. *Televerkets bil*

Garagespelet utspelar sig på ett $n \times n$ -rutnät som beskriver ett garage. I överkanten på kolumn m i rutnätet finns garageutfarten (EXIT). I garaget finns en massa bilar parkerade; varje bil tar upp två eller tre rutor bredvid varandra. Någonstans i kolumn m finns televerkets bil parkerad. Uppgiften är att flytta bilarna i garaget så att televerkets bil kan komma ut genom garageutfarten. Den bil som är parkerad på en rad kan bara röra sig på den raden och den bil som är parkerad i en kolumn kan bara röra sig i den kolumnen. Att köra en bil från en plats till en annan räknas som *en* förflyttning, oavsett hur många rutor den flyttas.



I exemplet är $n = 6$, $m = 3$, bil 1 på F1F2F3, bil 2 på E5F5, bil 3 på C6D6E6, bil 4 på D2D3D4, bil 5 på A1B1, bil 6 på A2A3A4, bil 7 på A5A6, bil 8 (som är *televerkets bil*) på B3C3. En kortaste lösning är följande åtta förflyttningar: bil 2 till B5C5, bil 1 till F4F5F6, bil 5 till E1F1, bil 6 till A1A2A3, bil 7 till A4A5, bil 3 till A6B6C6, bil 4 till D4D5D6, bil 8 ut genom garageporten.

(10p) Beskriv utförligt en algoritm som hittar den kortaste lösningen (så få förflyttningar som möjligt). Indata är talen n , m , ursprungsplaceringen för televerkets bil och för de övriga bilarna.

(4p) Hitta på en lämplig datastruktur för att hålla reda på tidigare uppnådda positioner i spelet. (En position i spelet beskrivs av var alla bilarna för tillfället befinner sig.) Är det lämpligt att använda ett Bloomfilter?

6. *Gammalt tema?*

Du har kommit på vad tentans tema är och vill se om det förekommit tidigare. Skriv en KMP-automat som söker efter **TELETEMA** i en textfil med extentor. (4p) Ange också den next-vektor som definierar automaten.

(3p) Beskriv hur Rabin-Karp kan användas för att söka i en text efter strängen ovan.

7. *Låda med telefonnummer*

Man vill implementera datastrukturen Låda med följande operationer: `put(telefonnummer)` som stoppar in ett telefonnummer i lådan, `exists(telefonnummer)` som undersöker om ett visst nummer finns med och `print()` som skriver ut allt i lådan.

Men det vore praktiskt om samma Låda gick att återanvända för att lagra andra typer av data, till exempel objekt som representerar en Atom eller en Bok.

(4p) Hur gör man för att se till att klassen Låda går att återanvända?