

2D1343, TENTAMEN I DATALOGI FÖR ELEKTRO
Lördagen den 6 april 2002 kl 14–19

Maxpoäng tenta+bonus = 50+7. Betygsgränser: 25 poäng ger trea, 35 ger fyra, 45 ger femma. Resultatet senast 27 april på Nadas anslagstavla. Hjälpmedel: Weiss eller annan algoritmbok.

1. Båtkappsegling

Havskappseglingen Gotland Runt 2002! Ett myller av båtar och frågor! Hjälp till att konstruera ett informationssystem där alla deltagare ska kunna ringa från sina mobiltelefoner och genom tonval välja bland menyer och information. Information hämtas givetvis från Gotland Runts hemsidor på Internet.

- (2p) Rita en skiss över de komponenter (mobiltelefoner, datorer, datornät, ...) som ingår i systemet.
- (3p) Beskriv också vilka protokoll och programvaror som är inblandade samt hur du tror att dessa fungerar.

2. Båtbogsering

Att bogsera båtar kan vara intressant. Har man en stark bogserbåt och många båtar kan de kopplas ihop likt ett binärt träd (bogserbåten direktkopplas till max två båtar som var och en direktkopplas till max två båtar osv).

- (5p) Formulera en *rekursiv* tanke för hur långt hela bogserläpet inklusive bogserbåten är! Du kan anta att varje båt med bogserlinor tar upp en längd av 25 meter och att en båt som inte bogserar någon annan tar 5 meter.

3. Båttuvning

I hamnområdet är det alltid kaos. I år finns därför en specialkonstruerad hamn och en sinnrik enkelriktad slinga runt ön Sandhamn. Dagen före start kommer båtar till en klassificeringsstation för att bli tilldelad en startgrupp innan de lägger till i hamnen. Det är dock viktigt att båtarna i hamnen under startdagen kan avgå i startgruppsordning!

- (5p) Beskriv utförligt en algoritm som gör detta möjligt! Hamnen fungerar som en *abstrakt stack* och slingan runt ön fungerar som en *abstrakt kö*. Till din hjälp har du dessutom hamnchefens båt.

4. Båtsortering

Under Gotland Runt vill organisatörerna snabbt och ofta kunna visa alla båtars placering och vill därför sortera sin databas lite då och då. Beskriv utförligt alla aspekter av de två algoritmerna insättningsortering och quicksort. Du kan anta att antalet båtar är omkring $500 \approx 2^9$ och att det är sträckan kvar till mål som är intressant i placeringslistorna. Du kan också anta att information om varje båt fortlöpande uppdateras i databasen (utan att du behöver bry dig om hur).

- (2p) Vilken algoritm är bäst i detta fall? Motivera!

5. *Båtflytt*

Under en seglingstävling vill varje båt hitta den snabbaste vägen till målet. Problemet är att en segelbåt inte kan segla hur som helst och att den seglar olika snabbt beroende på vindriktning och styrka. Antag att havet förenklat består av en massa jämnt fördelade punkter med information om vindstyrka, vindriktning och vilka punkter som finns runt om.

(8p) Beskriv **utförligt** en algoritm som på ett så effektivt sätt som möjligt tar reda på vilka punkter som ligger utefter den snabbaste seglingsvägen givet en startpunkt och en slutpunkt.

(2p) Båtgäaren är orolig att hans miljövänliga bottenfärg ska nötas bort och vill därför istället ta den väg som är kortast (d vs minst antal steg). Förklara utförligt vad som behöver ändras i din föregående algoritm.

6. *Båtsökning*

(6p) Gotland Runt har en databas med information om varje båt. Beskriv för var och en av nedanstående sökningsalgoritmer vilka *datastrukturer* som är lämpliga att använda till databasen, algoritmernas *fördelar och nackdelar, komplexitet* samt beräkna *antalet jämförelser* som görs vid sökning då databasen innehåller drygt $500 \approx 2^9$ poster. Algoritmer: linjärsökning, binärsökning, hashning.

7. *Båt detaljer*

(6p) Beskriv i detalj hur hashningen i föregående uppgift fungerar, hur mycket minne som går åt och vad du behöver ta hänsyn till om du söker på olika saker.

(2p) Hur kan du göra databasen mer effektiv om du i sökningen bara är intresserad av huruvida en båt finns med eller ej?

8. *Båtvård*

Seglingen är över och det är dags att ta hand om seglen. Hur lagras egentligen information om ett segel? Räcker ett tal med segelytan i m^2 eller ska det kanske vara kvadratfot?

(3p) Motivera varför det är bäst med en abstrakt datatyp för ett segel. Ange också

(2p) exempel på metoder med deras namn, in- och utdatatyper, tex
`public utdatatyp segelmetod(indatatyp variabelnamn)`