

Tentamen

dt046g/dt064g Datastrukturer och algoritmer

Martin Kjellqvist*

2016-04-18

Instruktioner

Läs igenom frågorna noggrant innan du börja besvara dem. Du har begränsat med tid, planera hur du ska besvara frågorna. Besvara endast det som efterfrågas. Skriv inte om saker som inte berörs av frågan.

Skriv svaren på erhållna svarspapper, inte på tentan. Varje ny fråga besvaras på ett nytt svarspapper. Skriv bara på en sida på svarspapperet.

Skriv tydligt. Om svaret är oläsligt får du 0 poäng - även om svaret är korrekt. Frågorna är *inte* ordnade efter svårighetsgrad.

Tid 5 timmar.

Hjälpmedel Inga.

Max poäng 50

Antal frågor 9

Preliminära gränser

$E \geq 40\%$, $D \geq 50\%$, $C \geq 60\%$, $B \geq 75\%$, $A \geq 90\%$.

Frågor

- (4p) 1. Vi har tagit upp några sorteringsalgoritmer med $O(N^2)$ karaktär i kursen. Både *insertion sort* och *selection sort* har egenskaper som i vissa situationer gör dem lämpliga kandidater. Beskriv dessa situationer.
- (3p) 2. Ordna följande uttryck efter asymptotisk tillväxttakt. Ange om uttrycket är ett polynom eller inte.
- $N^2 \log N$
 - 2^N
 - $N^2 + 3i\sqrt{N}$
- (5p) 3. Beskriv hur shellsort fungerar. Ange en övre gräns för dess komplexitet.

*martin.kjellqvist@miun.se

- (9p) 4. Följande problem och situationer har en tidskomplexitetskaraktär.

Din uppgift är att identifiera komplexiteten i situationen. Du behöver inte lösa problemen, de finns där för att illustrera en situation.

I en tabell anger du komplexiteten med $O()$ notation för varje situation, beskriv ditt resonemang.

Ordna tabellen så att den minst komplexa står först i tabellen, den mest komplexa sist.

- A I en nöjespark finns en stor planterad gränslabyrint. Du befinner dig utanför ingången till labyrinten. Du har blivit lovad att du kan hitta ut genom att i varje korsning hela tiden svänga vänster.

Den djupaste återvändsgränden är 2 km från ingången. Den närmaste återvändsgränden är 1000 m från ingången. Den närmaste vägen till utgången är 1200 meter. Var 30:e meter kommer du antingen till en trevägskorsning där du måste välja väg, eller en återvändsgränd.

Labyrinten är så konstruerad att man inte riskerar att gå i cirklar.

Hittar du utgången innan nöjesparken stänger för kvällen?

- B Din något excentriske granne, Klemetz, har bett dig att hjälpa till vid en provskjutning av hans nyinförskaffade haubits.

Hans haubits står uppställd på en avsides gård utanför Söråker. Hans målområde ligger en mil ut i havet.

Klemetz har kört ut Janssons brygga till målområdet och ankrat sin egen båt på behörigt avstånd från bryggan.

Du ska träffa Janssons brygga genom att ställa in riktning och elevation på haubitsen. Efter varje salva rapporterar Klemetz hur du träffat och du gör nödvändiga korrekitioner.

Det är en strålande fin och vindfri dag.

Träffa Janssons brygga.

(En haubits är en kanon som kan skjuta höga projektiler.)

- C Du har blivit antagen till "Fångarna på fortet".

Du blir ställd i en labyrint. För att klara dig ur labyrinten måste du passera N dörrar. Inför varje dörr måste du trycka på en vit eller svart knapp. Om du tryckt på fel knapp innan du går igenom dörren aktiveras en falllucka som bryskt transporterar dig till början av labyrinten. Dörrarna är konfigurerade på samma sätt vid nästa försök.

Ta dig ur labyrinten.

- (4p) 5. Beskriv RLE-komprimering.

För vad slags data är RLE en lämplig komprimeringsmetod?

Vad är maximal komprimeringsgrad för RLE-komprimering?

6. En graf som inte innehåller några cykler kan enkelt representeras i en trädstruktur.

- (2p) (a) Ange en algoritm som kan skapa denna trädstruktur.

- (4p) (b) Beskriv steg för steg en metod för att upptäcka cykler i en graf.

- (5p) 7. Vilka invarianter gäller för ett röd-svart träd.

- (6p) 8. Vilka blir delresultaten av att tillämpa en quicksort på tecknen

examquestion

Förklara delstegen. Din förklaring måste vara välstrukturerad och lättläst.

- (8p) 9. Implementera en komplett klass i c++ kod som representerar en kö. Stacken ska innehålla operationerna `enqueue(e)` och `dequeue():e`, med vanlig semantik.

Beskriv vilken komplexitet de båda operationerna har i din kod.

Ange vilken komplexitet push och pop har optimalt. Om din lösning inte har denna komplexitet beskriver hur man går tillväga för att uppnå den.

Din enda restriktion är att du inte får använda klassen `std::queue<T>`.

Lycka till,
Martin