

Tentamen Datastrukturer och algoritmer, dt046g, dt064g

Martin Kjellqvist

2010-03-15

Instruktioner

Inga hjälpmedel

Maximalt en uppgift per papper, du kan använda så många papper du önskar för varje uppgift. Skriv endast på en sida. Ange skrivkod på alla sidor. Om skrivkod inte finns tillgänglig skriver du ditt namn och personnummer på alla sidor. Skriv enkelt och tydligt för att undvika feltolkningar. Du får ej använda röd penna.

En uppgift med flera delar kan ge poäng utan att alla delar är lösta. Uppgifterna är inte ordnade efter svårighetsgrad. Läs igenom uppgifterna noggrant.

Preliminära gränser

A/B/C/D/E: 45/40/35/30/25 p

Max: 50 p

Lycka till!

/Martin

Uppgifter

1. Ordna följande uttryck efter deras tillväxttakt, långsammast växande först. Ange om uttrycket är ett polynom eller inte

N^2

$N + N + N$

$N \log N$

$\log(2^N)$

\sqrt{N}

(4 p)

2. Beskriv i vilken ordning noderna i en graf besöks av DFS algoritmen. Konstruera och beskriv ett exempel för DFS där BFS inte har samma besöksordning och peka ut var skillnaden uppträder. (5 p)
3. Beskriv hur en prioritetskö bör implementeras för att få $\log(n)$ beteende för både insättning och uthämtning. Använd c++ kod eller pseudokod. (6 p)
4. Du har fått i uppdrag att implementera en effektiv sorteringsalgoritm för speciella indata. Du har beslutat använda dig av en kombination av quicksort och insertionsort. Beskriv sorteringsalgoritmen (på vanlig algoritmisk form). Du ska inte hitta på en algoritm. Du ska beskriva en vanligt förekommande sorteringsmetod som beter sig på detta sätt. (5 p)
5. Utforma ett experiment där du ska undersöka hur du utformar den del av algoritmen som väljer huruvida quicksort eller insertionsort ska användas för att metoden i uppgift 4 ska vara optimerad för typiska indata. (5 p)
6. Hur fungerar ett röd-svart träd? Används strukturen i skarp kod? Beskriv insättning. Ange tidskomplexiteten för alla elementära operationer. (6 p)
7. Beskriv principen bakom en sk. xor-swap, dvs en implementation av swap som inte kräver en temporär lagringsyta. Ge exempel för några icke-triviala värden. (3 p)
8. Vilka blir delresultaten av att tillämpa en vanlig insertionsort på tecknen i strängen "examquestion"? Förklara delstegen. (6 p)

9. Vilken tidskomplexitet kommer följande kodavsnitt a med avseende på size? Process(A, N) är $O(1)$. Motivera.

```
function Complex(array[], size){  
    if ( size == 1 ) return 1;  
    if ( rand() / float(RAND_MAX) < 0.9 ){  
        result = Process(array, size*size);  
        result = result + Complex(array, size-1);  
        return result + Process(array, size*size);  
    }  
}
```

(6 p)

10. Implementera en komplett klass i c++ kod som representerar en sorterad lista heltal (implementationen behöver inte hålla en sorterad lista internt. Ta med de enklaste operationerna push(int) och get(int order):int. Order = 0 anger det minsta elementet, order = 1 anger det näst minsta elementet osv. Utifrån din implementation anger du komplexiteten hos de båda operationerna. (4 p)