

Tentamen

dt046g, dt064g Datastrukturer och algoritmer

Martin Kjellqvist*

2017-05-29

Instructions

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 svarsapper, inte på tentan. Varje ny fråga besvaras på ett nytt svarsapper. Skriv bara på en sida på svarsapperet.

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

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

Questions

- (4p) 1. Ordna följande uttryck efter deras tillväxttakt, långsammast växande först. Ange om uttrycket är ett polynom eller inte
- N^2
 - $N + N$
 - $N \log N$
 - \sqrt{N}

*martin.kjellqvist@miun.se

• 2^N

- (5p) 2. (a) Beskriv i vilken ordning noderna i en graf besöks av BFS algoritmen. Konstruera och beskriv ett exempel för DFS där BFS inte har samma besöksordning och peka ut var skillnaden uppträder.
- (2p) (b) Ange vilka datastrukturer som är lämpliga för att implementera respektive algoritmen.
- (3p) 3. 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.
4. Grafer: terminologi och principer. Ge korta beskrivningar.
- (2p) (a) Vad innebär det att en graf är riktad?
- (2p) (b) Hur implementerar man en riktad graf i en adjacency matrix.
- (2p) (c) Beskriv någon metod för att konstruera ett träd men en given nod som rot till trädet.
- (6p) 5. Vilka blir delresultaten av att tillämpa en vanlig mergesort på tecknen

examquestion

Förklara delstegen.

- (4p) 6. Beskriv vilka egenskaper som är önskvärda i en hashfunktion för att användas till att indexera en hashtabell. Ange hur dessa egenskaper kan uppnås.
- (3p) 7. Beskriv hur java har implementerat hashfunktionen för String-klassen.
8. Sortering med stackar. Du har följande datastruktur. Alla operationer är $O(1)$

```
1 class Stack
2 {
3 public:
4     Stack();
5     void Push(int i);
6     int Pop();
7     int Peek() const;
8     bool IsEmpty() const;
9 };
```

- (6p) (a) Du ska författa en algoritm som med hjälp av tre stackar S_{in} , S_{temp} , S_{out} sorterar indata S_{in} till utdata S_{out} . Det är en god ide att imitera någon känd sorteringsalgoritm som själv utnyttjar stackar.
- (1p) (b) Vilken sorteringsalgoritm imiterar du?
- (1p) (c) Vilken tidskomplexitet har den imiterade algoritmen?
- (3p) (d) Vilken tidskomplexitet har din algoritm? Motivera!
- (6p) 9. Implementera en komplett klass i c++ kod som representerar den stack som beskrivs i den tidigare uppgiften. Ta med samtliga beskrivna operationer. Utifrån din implementation anger du komplexiteten hos de båda operationerna.

Lycka till,
Martin