

Algoritmer, datastrukturer och komplexitet hösten 2015

Ommästarprov

Detta mästarprov är avsett för den som ännu inte är godkänd på mästarprov 1 eller mästarprov 2 (eller båda) samt den som har betyg E eller D på mästarprov 1 och vill ha betyg C. Mästarprovet ska lösas **individuellt** och redovisas både skriftligt och muntligt. Inget samarbete är tillåtet, se vidare hederskodexen. Du ska alltså inte diskutera lösningar med någon annan fram till dess att alla muntliga redovisningar är avklarade.

Du ska lämna in den **skriftliga lösningen** som en PDF-fil på kurswebben för adk15 senast 4 januari 2016 klockan 24.00 (och tidigast 29 december 2015).

Den **muntliga redovisningen** ska göras 7–8 januari 2016. Boka senast 4 januari 24.00 tid för muntlig redovisning på kurswebbsidan.

Det är viktigt att du förbereder dig inför den muntliga redovisningen. För att en uppgift ska godkännas ska du kunna förklara och motivera algoritmen eller reduktionen muntligt och reda ut eventuella oklarheter.

Läs uppgiften mycket noga så att du inte råkar basera dina lösningar på en missuppfattning. Fråga Stefan (snilsson@kth.se) eller Viggo (viggo@kth.se) om något är oklart.

1E. Algoritmer, för betyg E på mästarprov 1

Betygskriterium: utveckla algoritmer med datastrukturer för enkla problem givet en konstruktionsmetod. Denna uppgift ska göras endast av den som inte är godkänd på mästarprov 1!

Vi har fått en lista med n stycken ord, alla av samma längd m . Orden är skrivna med de engelska bokstäverna a till z .

Designa en algoritm som väljer ut alla ord i listan som matchar ett reguljärt uttryck bestående av m stycken $[]$ -uttryck. Ett $[]$ -uttryck matchar precis en bokstav som finns mellan hakarna; $[abc]$ matchar till exempel någon av bokstäverna a , b eller c . Det går också att skriva $[a-c]$, som matchar någon av bokstäverna från a till c .

Ett exempel på ett giltigt uttryck är $[a][a-z][x]$, som alltså beskriver alla strängar av längd 3 som börjar med a och slutar med x .

Eftersom $n \gg m$ så är det värt att bygga en datastruktur som gör det möjligt att snabbt matcha ett ord mot det reguljära uttrycket. Beskriv hur en sådan datastruktur kan se ut och beskriv i pseudokod hur man bygger den.

Beskriv också i pseudokod hur matchningen går till. Det räcker att du beskriver hur man matchar ett ord.

Analysera tidskomplexiteten för algoritmerna uttryckt i m och n .

1C. Algoritmer, för betyg C på mästarprov 1

Betygskriterium: utveckla algoritmer med datastrukturer för icke-triviala problem. Denna uppgift får göras endast av den som redan har betyg E eller D på mästarprov 1!

En kakelläggare har fått i uppdrag att lägga en list av kakel. Listen ska ha längden n och höjden 1. Det finns ett stort förråd av vita kakelplattor av storlek 1×1 , 1×2 och 1×4 . Gör en algoritm som räknar ut på hur många olika sätt det går att göra en sådan list. (En list av längd 4 kan göras på 6 olika sätt: $1 + 1 + 1 + 1$, $1 + 1 + 2$, $1 + 2 + 1$, $2 + 1 + 1$, $2 + 2$, 4 .)

Beskriv algoritmen med pseudokod och analysera dess tidskomplexitet uttryckt i n . Algoritmen måste vara polynomisk. Motivera också att algoritmen är korrekt.

2E. Komplexitet, för betyg E på mästarprov 2

Betygskriterium: förklara principerna, utföra enklare reduktioner mellan givna problem. Denna uppgift ska göras endast av den som inte är godkänd på mästarprov 2!

Den här gången har kakelläggaren fått en svårare uppgift som är en generalisering av problemet i uppgift 1C. Hon har tillgång till plattor av vilka heltalsdimensioner $1 \times n$ som helst och lagret av plattor är begränsat. Det kan till exempel finnas 6 plattor av storlek 1×1 , 2 plattor av storlek 1×3 , 5 plattor av storlek 1×12 och 1 platta av storlek 1×15 . Antalet plattor som ingår i listan spelar ingen roll.

Din uppgift är att först formulera detta problem som ett beslutsproblem genom att införa ett mål och en målfunktion, och sedan visa att detta problem är NP-fullständigt med hjälp av en reduktion av delmängdssumma.

Det är viktigt att du visar att du behärskar principerna för hur man visar att problem är NP-fullständiga.