Viggo Kann, KTH CSC

Teoritenta i Algoritmer (datastrukturer) och komplexitet för KTH DD1352/2D1352/DD2354/2D1354 och SU 2010-05-27 klockan 9.00–11.00 med efterföljande kamraträttning

Inga hjälpmedel är tillåtna. Skriv svaren direkt på blanketten. Skriv **inte** namn eller personnummer på tentan.

Bonuspoäng från läsåret 2009/2010 kan tillgodoräknas på denna tenta. 14 poäng krävs för betyg E (godkänt), 17 poäng för betyg D och 20 poäng för betyg C. Den får godkänt på tentan och har redovisat extralabben får extralabbens betyg som tentabetyg.

Lämna in tentan när du är klar, men tidigast 10.00. Lämna sedan salen, men återvänd klockan 11.10, för då tar rättningen vid. Varje tentand ska rätta en annan (anonym) tentands tenta. Därefter kontrollerar Viggo rättningen och för in resultaten i res ikväll.

1. (8 p) Är följande påståenden sanna eller falska? Ringa in rätt svar! För varje deluppgift ger riktigt svar 1 poäng och ett övertygande motiverat riktigt svar 2 poäng.

   a) Det existerar problem som är så svåra att inte någon algoritm kan lösa dom, oavsett hur lång tid algoritmen får på sig.

   sant **falskt**

   Motivering:

   b) En algoritm med tidskomplexiteten $O(\log(n!))$ går i polynomisk tid.

   sant **falskt**

   Motivering:

   c) Bloomfilter kan med fördel vara av tvåpotensstorlek, om man väljer rätt hashfunktioner.

   sant **falskt**

   Motivering:

   d) Om problemet A kan reduceras till problemet B så är B mindre och enklare än A, komplexitetsmässigt.

   sant **falskt**

   Motivering:

2. (3 p) Följande utsaga är sann: *Varje beslutsproblem vars ja-lösningar kan verifieras i polynomisk tid kan lösas av en algoritm som bara använder polynomiskt mycket minne.*

   a) Formulera utsagan med bara matematiska tecken och namn på komplexitetsklasser.

   b) Skissa ett bevis för att utsagan är sann.

3. (3 p) Totalsökning är en av de fyra viktigaste algoritmkonstruktionsmetoderna som nämnts i kursen. Vilka är de andra tre? Ge för varje konstruktionsmetod exempel på **en** algoritm från kursen som bygger på den metoden. Skriv antingen namnet på algoritmen eller ange både problemet som algoritmen löser och ett Θ-uttryck för tidskomplexiteten.

Konstruktionsmetod:	Exempel på algoritm:
....	....
....	....
....	....

4. (6 p) Vi söker i denna uppgift en polynomisk heuristik som ger en så bra lösning till TSP (handelsresandeproblemet) som möjligt, ju kortare tur desto bättre. Algoritmen ska också ge ett så bra mått som möjligt på hur långt bort från den optimala lösningen som den hittade lösningen som mest är, uttryckt i procent.

   Till ditt förfogande finns följande färdiga polynomiska algoritmer:

$\langle \Pi, s\rangle = \text{Christofides}(n, D)$	Approximationsalgoritm med approximationskvot 3/2.
$\langle \Pi, s\rangle = \text{NearestInsertion}(n, D)$	Deterministisk heuristik.
$\langle \Pi, s\rangle = \text{RandomInsertion}(n, D)$	Probabilistisk heuristik.
$\langle \Pi_2, s\rangle = \text{LocalSearch2Opt}(n, D, \Pi_1)$	Deterministisk heuristik som ger en förbättring av lösningen Π_1 med hjälp av lokal sökning.

   där Π beskriver en lösning som en permutation av städerna, s är lösningens (turens) längd, n är antalet städer och D är en symmetrisk $n \times n$-matris som beskriver avstånden mellan städerna. Anta att avstånden är positiva heltal och uppfyller triangelolikheten.

   Din algoritm ska ta n och D som indata och returnera trippeln $\langle \Pi, s, g\rangle$ som utdata, där g är måttet på hur långt algoritmens lösning som mest är från den optimala.