Försättsblad till skriftlig tentamen vid Linköpings universitet

Datum för tentamen	2019-08-27
Sal (1)	TER2(6)
Tid	14-19
Utb. kod	TDDD20
Modul	TEN1
Utb. kodnamn/benämning **Modulnamn/benämning**	Konstruktion och analys av algoritmer Skriftlig tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	5
Jour/Kursansvarig Ange vem som besöker salen	Peter Jonsson
Telefon under skrivtiden	013 282415, 070 7734389
Besöker salen ca klockan	15.30, 17.30
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Annelie Almquist 013 282934 annelie.almquist@liu.se
Tillåtna hjälpmedel	Kursboken "Introduction to Algorithms" (Cormen et al.) Ordlista
Övrigt	
Antal exemplar i påsen	

Linköpings Universitet
Institutionen för datavetenskap
Peter Jonsson

TDDD20 Konstruktion och analys av algoritmer

Tentamen tisdagen den 27:e augusti 2019, kl 14.00–19.00.

Hjälpmedel: Kursboken "Introduction to Algorithms" (Cormen, Leiserson, Rivest & Stein). Ordlista.

Poäng: Totalt kan 40 poäng erhållas. För godkänt krävs ca 16 poäng.

Jourhavande lärare: Peter Jonsson, tel 28 24 15 eller 070 773 43 89.

Allmänt: Skriv läsligt. Onödigt komplicerade lösningar kan leda till poängavdrag. Bristfälligt motiverade lösningar leder ofelbart till poängavdrag. Om inte annat anges skall presenterade algoritmer vara körbara på en processor av standardtyp (vilket utesluter exempelvis kvantdatorer och massiv parallellism). Slumpbitar får användas av algoritmer vid behov och de förutsätts kunna genereras i konstant tid.

Uppgifterna i tentamen är inte ordnade efter svårighetsgrad.

Lycka till!

Peter

Uppgift 1. (8p)
Konstruera en polynomisk algoritm som hittar största oberoende mängder i oriktade träd. Med andra ord, givet ett träd $T = (V, E)$, hitta en delmängd $V' \subseteq V$ med maximal storlek sådan att för alla $x, y \in V'$ så gäller att $\{x, y\} \notin E$.

Uppgift 2. (8p)
Antag att vi har en mängd av n intervall $[a_i, b_i]$ på tallinjen. Konstruera en algoritm som i $O(n \log n)$ tid beräknar den totala längd av tallinjen som täcks av intervallen. Observera att det inte räcker att addera intervallens längder $b_i - a_i$ eftersom intervallen kan överlappa varandra. Antag för enkelhets skull att intervallens start- och slutpunkter alltid är positiva heltal. Aritmetiska operationer mellan heltal får antas gå att beräkna i $O(1)$ tid.

Uppgift 3. (8p)
Konstruera en algoritm som löser följande problem i $O(n \cdot B)$ tid:

INDATA: två positiva heltal n och B samt en vektor $v[1...n]$ av positiva noll-skilda heltal.

UTDATA: en delmängd $S \subseteq \{1, \ldots, n\}$ som är så stor som möjligt och uppfyller $\sum_{i \in S} v[i] = B$. Om ingen delmängd uppfyller villkoret ska "nej" matas ut.

Aritmetiska operationer mellan heltal får antas gå att beräkna i $O(1)$ tid. Notera att problemet är NP-hårt men detta implicerar *inte* att P=NP eftersom det behövs högst $1 + \log_2 B$ bitar för att representera talet B.

Uppgift 4. (8p)
Betrakta följande grafproblem:

Indata: Sammanhängande graf G med n noder och m bågar.
Utdata: Alla par av noder x, y sådana att om man tar bort x och y ur G så blir den resulterande grafen ej sammanhängande.

Konstruera en algoritm som löser detta problem i $O(n \cdot m)$ tid. Observera att man kan kontrollera om en graf är sammanhängande i $O(m)$ tid. Alltså kan problemet trivialt lösas i $O(n^2 \cdot m)$ tid.

Uppgift 5. (8p)
En *Horn-klausul* är en klausul som innehåller maximalt en icke-negerad variabel; exempel är (x), $(\neg x)$, $(\neg x \vee \neg y \vee \neg z)$ och $(\neg x \vee \neg y \vee \neg z \vee w)$. En *Horn$^+$-formel* är en propositionslogisk formel $F = C_1 \wedge \ldots \wedge C_k$ där varje C_i, $1 \leq i \leq k$, antingen är en Horn-klausul eller av typen $(x \vee y)$ (d.v.s. innehåller två icke-negerade literaler). Visa att satisfierbarhetsproblemet för Horn$^+$-formler är NP-fullständigt.