# Försättsblad till skriftlig tentamen vid Linköpings universitet

| | |
|---|---|
| **Datum för tentamen** | 2019-06-11 |
| **Sal (1)** | TER2(12) |
| **Tid** | 8-13 |
| **Utb. kod** | TDDD20 |
| **Modul** | TEN1 |
| **Utb. kodnamn/benämning**<br>**Modulnamn/benämning** | Konstruktion och analys av algoritmer<br>Skriftlig tentamen |
| **Institution** | IDA |
| **Antal uppgifter som ingår i tentamen** | 5 |
| **Jour/Kursansvarig**<br>Ange vem som besöker salen | Peter Jonsson |
| **Telefon under skrivtiden** | 013-282415, 070-7734389 |
| **Besöker salen ca klockan** | Kl 9.15, 11.15 |
| **Kursadministratör/kontaktperson**<br>(namn + tfnr + mailaddress) | Annelie Almquist, tel 013-282934, annelie.almquist@liu.se |
| **Tillåtna hjälpmedel** | Kursboken " Introduction to Algorithms" ( Cormen, Leiserson, Rivest & Stein ). Ordlista. |
| **Övrigt** | |
| **Antal exemplar i påsen** | |

Linköpings Universitet
Institutionen för datavetenskap
Peter Jonsson

# TDDD20 Konstruktion och analys av algoritmer

Tentamen tisdagen den 11:e juni 2019, kl 8.00–13.00.

**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 013 282415, 070 7734389.

**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)
Låt $G = (V, A)$ vara en riktad graf utan självloopar (dvs det finns inga bågar $(v, v) \in A$). Låt funktionen $w : A \to \{1, 2, 3, \ldots\}$ vara en viktning av bågarna. Betrakta en väg

$$v_1 \xrightarrow{a_1} v_2 \xrightarrow{a_2} \ldots \xrightarrow{a_{k-2}} v_{k-1} \xrightarrow{a_{k-1}} v_k$$

som startar i $v_1 \in V$ och slutar i $v_k \in V$. Man säger att *vägens min-kapacitet* är $\min\{w(a_1), \ldots, w(a_k)\}$.

Konstruera en algoritm som givet två distinkta noder $v, w \in V$ och ett tröskelvärde $T$ avgör om det finns en väg mellan $v$ och $w$ med en min-kapacitet som är $T$ eller större. Algoritmen ska gå i högst $O(|V| + |A|)$ tid.

**Uppgift 2.** (8p)
Betrakta följande beräkningproblem:

**Indata:** En mängd intervall[1] $\mathcal{I} = \{I_1, \ldots, I_n\}$ på den rationella tallinjen där varje intervall $I_j$, $1 \leq j \leq n$, har en positiv och noll-skild vikt $w(I_j)$.
**Utdata:** En delmängd $\mathcal{I}'$ av $\mathcal{I}$ som uppfyller följande krav:

1. om $I, J \in \mathcal{I}'$ och $I \neq J$ så gäller att $I$ och $J$ är disjunkta, dvs $I \cap J = \emptyset$, samt
2. $\mathcal{I}'$ har största möjliga totalvikt, dvs. $\sum_{I \in \mathcal{I}'} w(I)$ är så stor som möjligt.

Konstruera en algoritm som löser detta problem i polynomisk tid.

**Uppgift 3.** (8p)
Vi har i kursen studerat *Horn-klausuler*, d.v.s. klausuler som innehåller högst en icke-negerad literal (t.ex. $(p)$, $(\neg p)$, $(\neg p \vee \neg q \vee \neg r)$ och $(\neg p \vee \neg q \vee r \vee \neg s)$ men inte $(\neg p \vee q \vee r)$.) En *Horn-formel* är en propositionslogisk formel på konjunktiv normalform som endast innehåller Horn-klausuler. Vi vet att satisfierbarhet för Horn-formler kan avgöras i polynomisk tid.

Visa att om man tillåter klausuler av typen $(p \vee q)$ att användas tillsammans med Horn-klausuler så blir satisfierbarhetsproblemet NP-fullständigt.

---

[1]Ett intervall $I$ representeras med ett par av tal $(I_s, I_f)$ där $I_s < I_f$. Antag att både $I_s$ och $I_f$ är positiva heltal.

**Uppgift 4.** (8p)
Maximeringsproblemet MAX CUT definieras så här:

**Indata:** Oriktad graf $G = (V, E)$ utan självloopar, dvs inga bågar går från nod $v$ till nod $v$.
**Lösning:** Två mängder $V_1, V_2$ sådana att $V = V_1 \cup V_2$ och $V_1 \cap V_2 = \emptyset$.
**Mått:** Antal bågar som har en ändpunkt i $V_1$ och en ändpunkt i $V_2$.

Konstruera en polynomisk algoritm som approximerar MAX CUT inom 2, d.v.s. hittar en lösning som har ett mått på minst 50% av den optimala lösningen. Algoritmen får vara randomiserad och då är kravet att den *förväntade* storleken på lösningen är 50% av optimum.

**Uppgift 5.** (8p)
Konstruera en algoritm som löser följande problem i $O(nB)$ 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 bara behövs $\log_2(B)$ bitar för att representera talet $B$.