

**Teoritentia i DD1352/DD2352 Algoritmer (datastrukturer) och komplexitet**  
**2017-08-16 kl 9.00-11.00**

Inga hjälpmedel är tillåtna. Skriv svaren direkt på blanketten.

Examinator: Viggo Kann, 08-7906292.

Bonuspoäng från läsåret 2016/2017 kan tillgodoräknas på denna tenta. För betyg E krävs 13 poäng för DD1352 och 12 poäng för DD2352. Den som dessutom klarar D-uppgiften får D och den som *dessutom* klarar C-uppgiften får C. Den som nästa vecka vill gå upp på munta för högre betyg (godkänt på skriftliga tentan krävs bl.a.) sätter ett kryss här:

Namn: ..... Personnr: .....

1. (6 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)  $n^2 \log n^2 \in \Omega(n^3)$ .

**sant      falskt**

Motivering:

b) Om man ska sortera en mängd teckensträngar av längd 5 så har radixsortering bättre tidskomplexitet än mergesortering.

**sant      falskt**

Motivering:

c) Om man ska implementera en prioritetskö utan att anropa biblioteksmetoder är det bättre att använda en heap än ett rödsvart träd (balanserat sökträd), även om tidskomplexiteten är densamma.

**sant      falskt**

Motivering:

2. (3 p) A, B, C och D är beslutsproblem. Anta att B är NP-fullständigt och att det finns kända polynomiska Karpreduktioner mellan problemen så här (en reduktion av A till B tecknas här  $A \rightarrow B$ ):

$$\begin{array}{ccc} A & \leftrightarrow & B \\ \uparrow & & \downarrow \\ C & \leftrightarrow & D \end{array}$$

Vad vet man då om komplexiteten för A, C och D? Sätt ett kryss i tabellen nedan för det man säkert vet och en ring för det som är möjligt men som man inte vet säkert.

	ligger i NP	är NP-fullständigt	är NP-svårt
A			
C			
D			

Namn: ..... Personnr: .....

3. a) (2 p) Definiera komplexitetsklassen  $NP$ .

b) (1 p) Definiera begreppet  $NP$ -fullständig. Du kan anta att komplexitetsklassen  $NP$  och begreppet Karpreduktion är kända.

4. (2 p) Vad kan man ha för nytta av en approximationsalgoritm? För vilka problem är det värdefullt att ha en approximationsalgoritm?

5. (Uppgift för betyg D, betygsriterium: *förklara principerna för hur man kan hantera problem med hög komplexitet*)

Vad innebär det att en approximationsalgoritm  $A$  approximerar ett optimeringsproblem inom faktorn  $q$ ? Vilken tidskomplexitet bör en approximationsalgoritm ha?

6. (Uppgift för betyg C, betygsriterium: *konstruera enkla heuristiker eller totalsökningsalgoritmer*)

Formulera pseudokod för en exponentiell algoritm som löser satisfierbarhetsproblemet CNF-SAT. Anta att det finns en metod  $Eval(\phi, \{x_1, \dots, x_n\})$  som evaluerar CNF-formeln  $\phi$  med dom booleska variabelvärdena  $x_1, \dots, x_n$  i linjär tid. Analysera tidskomplexiteten för din algoritm.

(Svara på ett separat papper.)