Examinator Viggo Kann viggo@kth.se

Teoritenta i DD2350 Algoritmer, datastrukturer och komplexitet 2021-04-09

Fyll i svaren direkt i detta formulär. Skriv inte namn eller personnummer på tentan. För varje uppgift finns länk till relevanta föreläsningar där teorin tagits upp.

För godkänt krävs dels 13 poäng, dels minst en halv poäng på både uppgift 1 och uppgift 2. Uppgift 1 och 2 examinerar lärandemålet *definiera och översätta centrala begrepp som P, NP, NP-fullständighet och oavgörbarhet.* För Fx krävs 11 poäng. Högre betyg äm godkänt ges inte.

Spara din ifyllda tenta som PDF-fil (kontrollera att den ser bra ut) och lämna in den i Peergrade **senast klockan 15.30**. Klockan 16.00 börjar den obligatoriska kamraträttningen i Zoom (länken finns också i examinationsrummet i Canvas). Varje tentand ska rätta en annan (anonym) tentands tenta.

Teoripoäng (bonus) från 2020 års kursomgång kan tillgodoräknas på denna teoritenta. Fyll i din teoripoäng från årets kurs (0-7 poäng, se Teoripoäng i Canvas):

Kryssa i att du läst och tänker följa regler för genomförande av teoritentan:

1. (1 p) [Nyckelbegrepp]

   a) Vad är den engelska termen för *simulerad härdning*?

   b) Vad är den svenska termen för *objective function*?

2. (1 p) [Föreläsning 22 och 23]
   Definiera nedanstående begrepp. Ge bara **en** definition av varje begrepp, inga exempel eller liknande. Definiera inte andra begrepp som ingår i dina definitioner.

   a) *NP*

   b) *oavgörbart*

3. (6 p) [Föreläsning 1, 25, 32 och 29]

Är följande påståenden sanna eller falska? Kryssa i rätt ruta för att svara. För varje deluppgift ger riktigt svar 1 poäng och ett *övertygande motiverat* (missa inte att förklara vad påståendet betyder) riktigt svar 2 poäng.

a) Låt $T(n) = 2 \cdot T(n/2) + 4n^2$ och T(1)=d för en konstant d. Då är $T(n) \in O(n \log n)$.

Svar: **sant** ☐ **falskt** ☐

Motivering:

b) SAT $\in$ PSPACE

Svar: **sant** ☐ **falskt** ☐

Motivering:

c) Innan man tillämpar heuristiken *lokalsökning* konstruerar man normalt en lösning med en konstruktionsheuristik.

Svar: **sant** ☐ **falskt** ☐

Motivering:

4. (1 p) [Föreläsning 30]

   Du försöker lösa en instans av ett NP-svårt maximeringsproblem. Anta att du kommit fram till en approximationsalgoritm med approximationskvot 3/2. Din instans har det optimala värdet 300.

   a) Vilket är det minsta värdet på lösningen din algoritm kan ge?

   b) Vilket är det största värdet på lösningen din algoritm kan ge?

5. (1 p) [Föreläsning 9]

   Anta att du formulerat lösningen till ett optimeringproblem som en rekursionsrelation. Förklara vad som gör att en algoritm som beräknar rekursionen med hjälp av dynamisk programmering går snabbare än en algoritm som använder en rekursiv funktion som implementerar rekursionsrelationen rakt av.

6. (3 p) [Föreläsning 24 och 22]

A, B, C, D, E, F och G är beslutsproblem. Anta att A är NP-fullständigt och att man känner till polynomiska Karpreduktioner mellan problemen så här (en reduktion av A till B tecknas här $A \to B$):

$$\begin{tikzcd}
 & A \arrow[dl] & B \arrow[l] & \\
C \arrow[dr] & & D \arrow[ul] \arrow[dl] & E \arrow[ul] \\
 & F \arrow[r] & G \arrow[ur] &
\end{tikzcd}$$

Anta i dessa frågor att P≠NP. Svara på frågorna genom att kryssa i motsvarande rutor.

a) Vilka av problemen kan vara NP-svåra?

B ☐ C ☐ D ☐ E ☐ F ☐ G ☐

b) Vilka av problemen måste tillhöra NP?

B ☐ C ☐ D ☐ E ☐ F ☐ G ☐

c) För vilka av problemen är det möjligt men inte säkert att dom är NP-svåra?

B ☐ C ☐ D ☐ E ☐ F ☐ G ☐

7. (1 p) [Föreläsning 20, 21 och 23]

Vilka av följande metoder kan (i princip) användas för att visa att ett problem A är tidskomplexitetsmässigt svårt?

Konstruera en polynomisk verifikationsalgoritm för att visa att A∈NP. ☐

Konstruera och analysera tidskomplexiteten för en algoritm för A. ☐

Visa en reduktion av ett känt svårt problem till A. ☐

Visa en reduktion av A till ett känt svårt problem. ☐

Visa att A är oavgörbart med ett motsägelsebevis. ☐