

Examinator Viggo Kann [viggo@kth.se](mailto:viggo@kth.se)

### Teoritenta i DD2350 Algoritmer, datastrukturer och komplexitet 2020-12-15

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 ges inte för DD2350. Den som tentar kursen DD1352 ska för betyg D och C göra separata extrauppgifter som finns i samma examinationsrum.

Spara din ifyllda tenta som PDF-fil (kontrollera att den ser bra ut) och lämna in den i Peergrade **senast klockan 10.30**. Klockan 11.00 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 årets kurs 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 *giriga algoritmer*?

b) Vad är den svenska termen för *computational model*?

2. (1 p) [[Föreläsning 1 och 24](#)]

Definiera nedanstående begrepp. Ge bara definitionen av varje begrepp, inga exempel eller liknande. Definiera inte andra begrepp som ingår i dina definitioner.

a) *bitkostnad* (vid analys av tidskomplexitet)

b) *NP-svårt problem*

3. (8 p) [Föreläsning [24](#), [20](#), [32](#), [1](#), [3](#) och [19](#)]

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* riktigt svar 2 poäng.

a) Om Turingreduktion skulle användas istället för Karpreduktion i definitionen av NP-svår skulle alla co-NP-svåra problem också vara NP-svåra.

Svar: **sant**                      **falskt**

Motivering:

b) Låt  $T(n) = 2 \cdot T(n/2) + 4n$  och  $T(1)=k$  för en konstant  $k$ . Då är  $T(n) \in O(n)$ .

Svar: **sant**                      **falskt**

Motivering:

- c) Värsta fallet för antalet filpositioneringar vid sökning i en hashtabell med krocklistor som ligger på fil är linjärt i antalet element eftersom alla element i värsta fallet kan hamna på samma index i hashtabellen. Värsta fallet för antalet filpositioneringar vid sökning i latmanshashning är också linjärt.

Svar: **sant**                      **falskt**

Motivering:

- d) Anta att du har en probabilistisk algoritm som givet en boolesk formel med  $n$  variabler försöker att avgöra om den är satisfierbar genom att upprepade gånger ( $M$  stycken, där  $M$  är ett stort tal) slumpmässigt välja värden på alla ingående variabler och testa om formeln satisfieras (och om ingen av dom  $M$  variabeltilldelningarna satisfierar formeln svarar att formeln inte är satisfierbar).

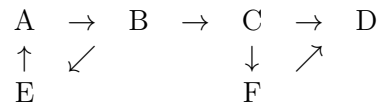
Den beskrivna algoritmen är en Las Vegas-algoritm.

Svar: **sant**                      **falskt**

Motivering:

4. (3 p) [Föreläsning 24 och 22]

A, B, C, D, E och F är beslutsproblem. Anta att A tillhör komplexitetsklassen P (OBS!) och att man känner till polynomiska Karpreduktioner mellan problemen så här (en reduktion av A till B tecknas här  $A \rightarrow B$ ):



Anta i dessa frågor att  $P \neq NP$ . Svara på frågorna genom att kryssa i motsvarande rutor.

a) Vilka av problemen kan vara NP-svåra?

A                      B                      C                      D                      E                      F

b) Vilka av problemen måste tillhöra NP?

A                      B                      C                      D                      E                      F

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

A                      B                      C                      D                      E                      F

5. (1 p) [Föreläsning 30]

Anta att X är ett optimeringsproblem (taget från verkligheten) som man skulle vilja lösa. Problemet formulerat som beslutsproblem är dock NP-fullständigt.

Ange fyra angreppssätt som tagits upp i kursen som kan göra att man ändå får fram tillfredsställande lösningar till det verkliga problemet.

En kursenkät skickas snart ut till alla som är registrerade på kursen. Vi hoppas att du svarar på den enkäten. Studentsynpunkterna är mycket viktiga. Vi läser och reflekterar över varenda synpunkt som lämnas. Tack på förhand!

*Viggo och Stefan*