

**Teoritenta i Algoritmer (datastrukturer) och komplexitet
för KTH DD1352–2354 2009-12-18 klockan 14.00–17.00**

Inga hjälpmedel är tillåtna. 10 poäng krävs för betyg E, 13 poäng för betyg D och 16 poäng för betyg C.

1. (8 p) Är följande påståenden sanna eller falska? För varje deluppgift ger riktigt svar 1 poäng och ett övertygande motiverat riktigt svar 2 poäng.

- a) Det går att multiplicera två n -bitstal i tid snabbare än $O(n^2)$.
- b) Dynamisk programmering kan användas för att hitta kortaste vägar i grafer.
- c) Det finns NP-problem som inte ligger i klassen PSPACE.
- d) Att avgöra om en graf kan 2-färgas är ett NP-fullständigt problem.

2. (3 p)

- (a) Beskriv Dijkstras algoritm för att hitta kortaste vägar i grafer.
- (b) Beskriv Kruskals eller Prims algoritm för att hitta minimala spännande träd i grafer.
- (c) Vilka av dessa algoritmer klarar av både positiva och negativa kantvikter som indata.
Ge motivering.

3. (2p) Problemet EKVIVALENTA TURINGMASKINER är problemet att givet två Turingmaskiner M_1 och M_2 avgöra om de uppträder likadant på alla indata x . Mer exakt menas att för varje indata x gäller att båda inte stannar eller att de stannar med $M_1(x) = M_2(x)$ d.v.s. med samma utdata.

Visa att detta problem är oavgörbart.

4. (3p)

Problemet PARTITIONERING är att givet en sekvens av heltal $x_1, x_2, \dots, x_n$ avgöra om de gör att dela upp i två disjunkta delar så att delarna ger samma summa. Problemet LÅDPACKNING är att givet reella tal $y_1, y_2, \dots, y_n$ som alla är större än 0 och ett heltal K , avgöra om talen kan packas i K lådor av storlek 1.

Visa hur problemet PARTITIONERING kan reduceras till LÅDPACKNING.

5. (4p)

- (a) Beskriv vad en approximationsalgoritm är. Ge exempel på en sådan algoritm.
- (b) Som vi har sett går inte HANDELSRESANDEPROBLEMET i sin mest generella form att approximera. Men antag att vi har tal $0 < \alpha < \beta$ och en speciell klass av kompletta grafer där som är alla kantvikter w uppfyller $\alpha < w < \beta$. Visa att det då går att konstruera en approximationsalgoritm med approximationskvot $B = \frac{\beta}{\alpha}$.