

Tentamen Algoritmer och Datastrukturer

Tentamensinformation

Tentamen består av två delar. Del I innehåller tio uppgifter för betyget E. Del II, som inte är obligatorisk, innehåller fyra uppgifter för betygen A-D. Maximal skrivtid är fyra timmar.

Varje lösning kommer att bedömas med 0-10 poäng.

Kravet för godkänt (E) är 50 poäng på del I. Du måste bli godkänd på del I för att kunna erhålla högre betyg.

Under förutsättning att du är godkänd på del I gäller följande betygsgränser för del II:

- Betyg D: 5-10 poäng
- Betyg C: 11-20 poäng
- Betyg B: 21-30 poäng
- Betyg A: 31-40 poäng

Lösningar anslås på kursens hemsida.

Uppgifter del I

Följande uppgifter avser betyget E. De är ordnade efter kursens olika delmoment. De är inte ordnade i svårighetsgrad. Krav på *fullständiga lösningar med motiveringar* gäller för samtliga uppgifter.

Exempel på en e-uppgift:

Beskriv en algoritm som avgör om en oriktad graf innehåller cykler eller inte. Algoritmen ska vara så snabb som möjligt.

Uppgifter del II

Följande uppgifter avser betygen A-D. De är ordnade efter kursens olika delmoment. De är inte ordnade i svårighetsgrad. Krav på *fullständiga lösningar med motiveringar* gäller för samtliga uppgifter.

Exempel på en möjlig högrebetygsuppgift:

Vi har en riktad graf G . Vi antar att kanterna i grafen är färgade blå och röda. Låt s vara en given startnod. Vi kallar en stig som startar i s för *alternerande* om den börjar med en blå kant och sedan har varannan kant röd och varannan blå. (Blå, röd, blå är alternerande. Stigen som bara består av en blå kant är alternerande. Vi säger för enkelhets skull att den "tomma stigen" utan kanter är alternerande.)

Låt $\text{Alt}(s)$ vara mängden av noder som kan nås med orienterade, alternerande stigar som börjar i s . Konstruera en algoritm som beräknar $\text{Alt}(s)$. Den ska ha så låg tidskomplexitet som möjligt. Ge algoritmen i lämplig kod. Förklara varför algoritmen är korrekt och gör en komplexitetsanalys.