Eksamen i fag
TDT4120 Algoritmer og Datastrukturer
Tirsdag 3. August 2004, kl 0900-1500

Faglig kontakt under eksamen: Magnus Lie Hetland tlf. 91851949
Hjelpemiddel: Alle kalkulatortyper tillatt. Alle trykte og håndskrevne hjelpemidler tillatt. Skriv svarene i de oppgitte rutene. Et svar uten begrunnelse teller ikke. En kan ta i bruk tilleggsark dersom dette er nødvendig. Før på ”Student nr.” på hvert svarark.

Oppgavene er merket #n/p, der n er oppgavenummeret og p er antall poeng som maksimalt kan oppnås når både svaret (Ja /Nei) og tilhørende begrunnelse er riktig.

Det kan maksimalt oppnås 50 poeng for besvarelsen.

#1/1: ”Best case” kjøretid for INSERTION SORT ved sortering av n elementer er O(n).

Svar:	Begrunnelse:

#2/3: Ved å bruke Master-teoremet finner en løsningen $T(n) = \Theta(n \log n)$ på rekurrensen $T(n) = 3\,T(n/3) + \log n$

Svar:	Begrunnelse:

#3/3: For et vilkårlig binært søketre med n noder kan vi skrive ut nodene i sortert rekkefølge på O(n) tid.

Svar:	Begrunnelse:

#4/1: Ethvert binært søketre med n noder har høyde O(logn).

Svar:	Begrunnelse:

#5/1: Enhver haug (heap) som benyttes av HEAPSORT for å sortere n elementer har høyde O(log n)

Svar: Begrunnelse:

#6/2: Haugen i HEAPSORT er tilfeldig ordnet.

Svar: Begrunnelse:

#7/2: Det er slik at $nlogn^2 = O(n^2)$.

Svar: Begrunnelse:

#8/3: MERGESORT bruker i worst-case $O(n^2)$ tid

Svar: Begrunnelse:

#9/2: En bredde-først (bredth first) søke-algoritme gjør bruk av en stakk.

Svar: Begrunnelse:

#10/2: En dybde-først (depth-first) søke-algoritme gjør bruk av en stakk.

Svar: Begrunnelse:

#11/3: En maksimal-matching i en bipartitt graf kan finnes ved Lineær-Programmering.

Svar: Begrunnelse:

#12/3: Hvis noen kantvekter i en rettet graf G = (V,E) er negative, kan den korteste veien fra node s til node t finnes ved å bruke Dijkstra's algoritme dersom vi først legger til en stor konstant C til alle E's kantlengder slik at alle disse blir ikke-negative.

Svar: Begrunnelse:

#13/1: Enhver DAG (directed acyclic graph) kan på en entydig måte sorteres topologisk.

Svar: Begrunnelse:

#14/1: Dijkstras algoritme er et eksempel på en grådighetsalgoritme.

Svar: Begrunnelse:

#15/1: Dijkstras algorime er et eksempel på dynamisk programmering.

Svar: Begrunnelse:

#16/3: Hvis alle kant-kapasiteter til en flyt-graf er et multiplum av 5, da vil også den maksimale flyten være dette.

Svar: Begrunnelse:

#17/4: I en flyt-graf med noder s,a,b,c,d,t lar vi f/m bety at f enheter flyter på en kant med kapasitet m. Vi har gitt følgende kantflyt: (s,a):2/4, (s,c):6/6, (c,a):3/7, (a,b):5/5, (b,c):0/2, (c,d):3/3, (b,d):1/4, (d,t):4/4, (b,t):4/6. Denne kantflyten er maksimal.

Svar: Begrunnelse:

#18/3: La P være den korteste veien fra s til t i en graf G=(V,E). Dersom vi øker lengden på alle kanter i E med 1, så vil P fortsatt være den korteste veien fra s til t.

Svar: Begrunnelse:

#19/3: Et dybde-først-søk i en graf er asymptotisk raskere enn et bredde-først-søk.

Svar: Begrunnelse:

#20/2: n heltall i området $[0,n^{100}]$ kan sorteres i lineær tid.

Svar: Begrunnelse:

#21/2: Grafen G er asyklisk hvis det ikke oppstår ”back-edges” under dybde-først-traversering av G.

Svar: Begrunnelse:

#22/4: Vi skal flettesortere (merge) k sorterte lister, hver med n/k elementer, ved følgende metode:

Flett de 2 første listene, flett så resultatet med den tredje listen, flett dette resultatet videre med den fjerde listen, og så videre, inntil den siste listen på n/k elementer blir flettet inn.

Påstanden er her at denne algoritmen krever Θ (kn) tid.

Svar: Begrunnelse: