KONTINUASJONSEKSAMEN I FAG 45011
ALGORITMER OG DATASTRUKTURER

Mandag 9. august 1993 kl. 0900-1300

Faglig kontakt under eksamen: Arne Halaas, tlf. 3442

Alle trykte og håndskrevne hjelpemidler tillatt.
Godkjent lommekalkulator tillatt.

Merk: Alle svar skal skrives i de angitte feltene på oppgavearket. Korte og presise svar honoreres best.

Oppgave 1

Gitt funksjonen

```
        function pow(x, n : integer) : integer:
        begin
(1)        if n = 0 then
(2)            pow := 1
           else
(3)        if n = 1 then
(4)            pow := x
           else
(5)        if even(n)  then
(6)            pow := pow(x ∗ x,n div 2)
           else
(7)            pow := pow(x ∗ x,n div 2) ∗ x:
        end;
```

(a) forklar kort hva pow kan benyttes til

(b) Er noen av programlinjene i pow overflødige?

(c) Hva vil effekten være ved å endre linje 7 til:
    (7a) pow := pow(x,n-1) ∗ x; ?

(d) Kun en av programlinjene nedenfor kan erstatte programlinje 6.
Forklar hvorfor.
    (6a) pow := pow(pow(x,2), n div 2)
    (6b) pow := pow(pow(x, n div 2), 2)
    (6c) pow := pow(x, n div 2) ∗ pow(x, n div 2)

(e) Angi/beregn tidskompleksiteten av pow, som gitt først i oppgaveteksten. Bruk O-notasjonen.

(f) Angi/beregn tidskompleksiteten av pow her med linje 7 erstattet av linje (7a), som gitt i deloppgave (c). Bruk O-notasjonen.

(g) Angi/beregn tidskompleksiteten av pow, nå med linje 6 erstattet av den eneste mulige kandidat blant (6a), (6b), (6c), som gitt i deloppgave (d). Bruk O-notasjonen.

Oppgave 2

Vi skal her se på korteste-vei-problemer. La G = (V,E) være en vektet og rettet graf med kilde s, og la G som i Dijkstras algoritme være initialisert ved INTIALIZE-SINGLE-SOURCE(G,s).

(a) Vis at dersom en etter et antall kall av RELAX(u,v,w) setter pekeren π(s) til en verdi forskjellig fra NIL, da inneholder G en sykel med negativ lengde (= samlet vekt).

(b) Forklar hva resultatet vil være ved å kjøre h.h.v. Dijkstras og Bellman-Fords algoritmer på en graf som inneholder en sykel med negativ lengde.

Oppgave 3

Vi definerer (kant-) koplingsgraden til en urettet graf G = (V,E) til å være k = "det minimale antall kanter som må fjernes for at G skal bli usammenhengende."

Eksempler :

(I) k = 1 når G er en trestruktur.
(II) k = 2 når G er en syklisk kjede.

Vis hvordan k kan finnes ved å kjøre en maks-flyt algoritme på høyst $|V|$ flytnettverk, hvert med O(V) noder og O(E) kanter.