

NTNU
Norges teknisk-naturvitenskapelige
universitet

Fakultetet for informasjonsteknologi,
matematikk og elektroteknikk

Institutt for datateknikk og
informasjonsvitenskap

BOKMÅL



Løsningsskisse til kontinuasjonseksamen i TDT4105
Informasjonsteknologi, grunnkurs
Onsdag 10. august 2011
9.00 – 13.00

Hjelpemidler: C

Tilleggshefte I, "Introduksjon til HTML, CSS, JSP og MYSQL"

Tilleggshefte II, "Introduksjon til: MATLAB"

Bestemt, enkel kalkulator: HP 30S eller Citizen SR270-X

Oppgave 1: Flervalgsoppgave (25 %)

Bruk de to vedlagte svarskjemaene for å svare på denne oppgaven (ta vare på det ene selv). Du kan få nytt ark av eksamensvaktene dersom du trenger dette. Kun ett svar er helt riktig. For hvert spørsmål gir korrekt avkryssing 1 poeng. Feil avkryssing eller mer enn ett kryss gir $-1/2$ poeng. Blankt svar gir 0 poeng. Du får ikke mindre enn 0 poeng totalt på denne oppgaven. Der det er spesielle uttrykk står den engelske oversettelsen i parentes.

1. Hva er en *tjener* i en klient-/tjenerarkitektur?

- a) **En tjener utfører tjenester på vegne av en klient.**
- b) En tjener etterspør tjenester fra klienter.
- c) En tjener er et program som vil ha noe utført på en annen maskin.
- d) Ingen av alternativene i a, b og c.

2. En del av et program består av en FOR-løkke, inne i en FOR-løkke, inne i en FOR-løkke. Alle løkkene løper gjennom tallene fra 1 til N. Tidsforbruket for denne programdelen vil være proporsjonal med:

- a) N^3
- b) $3N$
- c) $3N \log N$

3. Hva er hovedoppgaven til en *navnetjener*?

- a) **Å oversette domenenavn til IP-adresser.**
- b) Å oversette IP-adresser til domenenavn.
- c) Å holde rede på definerte variabelnavn når et program kjører.
- d) Ingen av alternativene i a, b og c.

4. Kari har utviklet et programsystem i den bedriften der hun har vært ansatt. Kan hun ta med seg og videreutvikle denne programvaren hvis hun slutter i bedriften?

- a) Ja, i følge Åndsverksloven er det programmereren som har rettighetene til programvaren.
- b) Nei, det er bedriften som eier programvaren.**
- c) Det er uavklart siden dette er ikke regulert av lovgiving. Rettighetene til programvaren må avtales mellom partene i hvert enkelt tilfelle.

5. Svart-boks testing er:

- a) Testing der man prøver ut systemets eller modulens indre oppbygging.
- b) Testing der man ser bort fra systemets eller modulens indre oppbygning og kun tar i betraktning de ytre egenskapene.**
- c) Testing der man ser bort fra systemenes eller modulenes indre oppbygning og prøver ut hvordan de ulike delene i et system fungerer sammen.

6. Hva er riktig rekkefølge når ulike minnetyper sorteres etter synkende hastighet?

- a) Register, cache, primærminne, sekundærminne.**
- b) Cache, registre, primærminne, sekundærminne.
- c) Primærminne, cache, sekundærminne, registre.

7. Når kan man *ikke* benytte seg av binærsøkingsalgoritmen?

- a) Når datamengden er så stor at det vil ta for lang tid.
- b) Når det er stor sannsynlighet for at det man søker etter ikke finnes i datamengden.
- c) Når datamengden ikke er sortert.**
- d) Man kan alltid bruke binærsøkingsalgoritmen.

8. Hva er i lovverket ikke definert som *sensitive* personopplysninger:

- a) Etnisk bakgrunn.
- b) Politisk oppfatning.
- c) Filosofisk oppfatning.
- d) Alle alternativene i a, b og c er sensitive personopplysninger.**

9. Vi skal kode tegnene a-å, A-Å, 0-9, mellomrom, spørsmålstegn, utropstegn, punktum, kolon, semikolon og bindestrek. Hvor mange binære siffer (engelsk: bits) er nødvendig i kodene når alle tegn skal representeres med like mange binære siffer?

- a) 6
- b) 7**
- c) 8

10. Hva blir resultatet når 81 kodes binært (i 2-tallsystemet)?

- a) 1011011
- b) 1010101
- c) **1010001**
- d) Ingen av alternativene i a, b og c er riktige.

11. Hva tilsvarer ABCD i det heksadesimale tallsystemet (16-tallsystemet) i 10-tallsystemet?

- a) 42561
- b) **43981**
- c) 44981
- d) Ingen av alternativene i a, b og c er riktige.

12. QoS er forkortelse for:

- a) Quantity of Storage.
- b) Quality of Servers.
- c) **Quality of Service.**
- d) Ingen av alternativene i a, b og c er riktige.

13. Vi har en sortert liste med 100 000 000 elementer. Ved binærsøking i denne listen, hvor mange sammenligninger må vi i verste fall gjøre?

- a) Omtrent 20.
- b) **Omtrent 27.**
- c) Omtrent 34.

14. `<p>...</p>` i en HTML-fil definerer:

- a) **Et avsnitt.**
- b) Et adressefelt.
- c) En hyperlenke.
- d) Ingen av alternativene i a, b og c.

15. $A \parallel \sim(B \ \&\& \ C)$ er usant (engelsk: false) når:

- a) **A false, B true, C true**
- b) A true, B true, C false
- c) A false, B false, C true
- d) Uttrykket er ikke usant for noen av alternativene i a, b og c.

16. Hva er *entropi*?

- a) **Et mål på informasjonsinnholdet i en melding.**
- b) En algoritme for effektiv koding av symboler.
- c) En feiltilstand som gjør at lesehodet på en harddisk ikke klarer å følge sporene nøyaktig og derfor blander data fra flere spor.
- d) Ingen av alternativene i a, b og c er riktige.

17. En URL (Uniform Resource Locator) har følgende format:

- a) **Protokoll, maskin, sti, fil**
- b) Protokoll, maskin, sti
- c) Domene, sti, fil
- d) Ingen av alternativene i a, b og c er riktige.

18. Anta at karakterene har følgende fordeling: A (12 %), B (24 %), C (35 %), D (20 %), E (7 %) og F (2 %). Hva er en korrekt Huffmankoding for A-F?

- a) **A: 001 B: 10 C: 11 D: 01 E: 0001 F: 0000**
- b) A: 010 B: 01 C: 10 D: 111 E: 1110 F: 1111
- c) A: 1111 B: 11 C: 1 D: 111 E: 11111 F: 111111
- d) Ingen av alternativene i a, b og c.

19. En IP-adresse er:

- a) **En numerisk adresse som identifiserer en datamaskin eller annen type enhet i et IP-nettverk.**
- b) En mekanisme som gjør det mulig å kontakte en Internet Provider, for eksempel når det oppstår feil i nettet.
- c) Et felt som identifiserer data i en relasjonsdatabase.

20. Hva er *inspeksjon* av programvare?

- a) Kjøring av programmer for å se om programvaren oppfyller kravene til systemet.
- b) Overvåking av variablenes verdier under programkjøring for å finne årsakene til feil.
- c) **Gjennomgang av programmer uten at de blir kjørt, med tanke på å finne feil og å forvise seg om at de er riktige.**

Oppgave 2: Programforståelse (10 %)

10201

Oppgave 3: Programmering (15 %)

a) Rente-funksjon:

```
function r = rente(belop, renteFot)
% rentefot i prosent

r = belop*renteFot/100;

end % rente
```

b) Terminbeløp-funksjon:

```
function tb = termBelop(laanBelop, renteFot, antallTerm)
% renteFot i %

r = renteFot/100;

tb = laanBelop*(1+r)*...
    (1 - 1/(1+r))/(1 - (1/((1+r)^antallTerm)));

end % termbelop
```

c) Restlån-funksjon:

```
function saldo = restLaan(laanBelop, renteFot, antTerminer,
termin)

tb = termBelop(laanBelop, renteFot, antTerminer);

saldo = laanBelop;

for i = 1:1:termin
    termRente = rente(saldo, renteFot);
    saldo = saldo - (tb - termRente)
end

if saldo < 0
    saldo = 0;
end

end % restLaan
```

Oppgave 4: Programmering (50 %)

a) (5%) Skriv funksjonen *random_picture* som tar inn parameteren *picsize*, og returnerer en kvadratisk tabell av tall på størrelse *picsize* som inneholder tilfeldige gråtoner (tall fra 0 til 255). Tabellen som er vist i figur 1 har *picsize* 6 etter som tabellen er en 6x6 matrise.

```
function [ picture ] = random_picture (picsize)
    for i = 1:1:picsize
        for j = 1:1:picsize
            picture (i,j) = floor(rand*256);
        end
    end
end
```

b) (5%) Skriv funksjonen *test_picture* som tar inn en todimensjonal tabell av tall, *table*. Funksjonen skal returnere sannhetsverdien *true* hvis tabellen er kvadratisk samt at tabellen ikke inneholder noen ulovlige tallverdier.

```

function [ result ] = test_picture (table)
    result = true;
    tab_size = size(table);
    if (tab_size(1) ~= tab_size(2))
        result = false;
    else
        for i=1:1:tab_size(1)
            for j=1:1:tab_size(2)
                if (table(i,j)<0 || table(i,j)>255)
                    result = false;
                end
            end
        end
    end
end

```

- c) (10%) Skriv funksjonen *filter_black_and_white* som tar inn en todimensjonal tabell av tall, *table*, samt et tall, *threshold*. Funksjonen skal returnere en endret tabell som er en svart/hvitt versjon av bildet representert i *table*. Alle verdiene i tabellen som er lavere enn *threshold* skal få verdien 0 (svart), mens alle andre verdier skal få verdien 255 (hvit). Kalles funksjonen *filter_black_and_white* på tabellen fra figur 1 med *threshold* satt til 127, gir det resultatet som vist i figur 2.

```

function [ table ] = filter_black_and_white (table,
threshold)
    table_size = size(table);
    for i = 1:1:table_size(1)
        for j = 1:1:table_size(2)
            if (table(i,j)<threshold)
                table(i,j) = 0;
            else
                table(i,j) = 255;
            end
        end
    end
end

```

- d) (10%) Skriv funksjonen *filter_inverse*, som tar inn en todimensjonal tabell av tall, *table*, og returnerer en tabell der innholdet i tabellen er invertert, dvs. alt som er lyst på bilde skal bli mørkt og motsatt. Figur 3 viser resultatet etter å ha kjørt *filter_inverse* på tabellen fra figur 1.

```

function [ table ] = filter_invers (table, threshold)
    table_size = size(table);
    for i = 1:1:table_size(1)
        for j = 1:1:table_size(2)
            table(i,j) = 255-table(i,j);
        end
    end
end

```

- e) (10%) Skriv funksjonen *histogram* som tar inn en todimensjonal tabell av tall, *table*, og returnerer en endimensjonal tabell som inneholder antall punkter i bildet (*table*) som har en bestemt gråtone. For bildet i figur 3 vil histogrammet for eksempel vise at bildet har 1 punkt med verdi 1, 1 punkt med verdi 3, 2 punkter med verdi 8, 2 punkter med verdi 15, 2 punkter med verdi 17 osv.

f)

```
function [ hist_table ] = make_histogram(table)
    hist_table = zeros(1,256);
    for i= 1:size(table)(1)
        for j= 1:size(table)(2)
            hist_table(table(i,j)+1)++;
        end
    end
end
```

- g) (10%) Bruk funksjonene ovenfor (selv om du ikke har løst deloppgavene) til å lage et Matlab-script som gjør følgende:

```
table = random_picture(40);
if(test_picture(table))
    table = filter_invers(table);
    hist_tab = make_histogram(table);
    bar(hist_tab);
    treshold=input('Treshold på b/w filter: ');
    table = filter_black_and_white(table,treshold);
    imshow(table,[0,255]);
end
```

Svarskjema flervalgsoppgave

Studentnummer: _____

Program: _____

Fagkode: _____

Dato: _____

Antall sider: _____

Side: _____

<i>Oppgave nr.</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>
1.1				
1.2				
1.3				
1.4				
1.5				
1.6				
1.7				
1.8				
1.9				
1.10				
1.11				
1.12				
1.13				
1.14				
1.15				
1.16				
1.17				
1.18				
1.19				
1.20				