

NTNU
Norges teknisk-naturvitenskapelige
universitet

Fakultetet for informasjonsteknologi,
matematikk og elektroteknikk

Institutt for datateknikk og
informasjonsvitenskap

BOKMÅL



Løsningsskisse til avsluttende eksamen i TDT4105
Informasjonsteknologi, grunnkurs
Torsdag 8. desember 2011
9:00 – 13:00

Hjelpemidler: C

Typegodkjent kalkulator: HP30S

Oppgave 1: Flervalgsoppgave (25 %)

- 1) **Hva er et hovedkort (motherboard)?**
 - a) **Et kretskort i en datamaskin der enheter som CPU, RAM, og andre enheter kobles sammen.**
 - b) En minnekrets som tar vare på systemets innstillinger.
 - c) En prosessor for tynne mobiltelefoner.
 - d) Bunnplata i et PC kabinett.
- 2) **Hva er en pakke (packet) i nettverkssammenheng?**
 - a) **En datablokk av fast lengde som sendes gjennom nettverket, fra avsender til mottaker.**
 - b) En datamelding som har ankommet og som står i kø for å bli levert til mottakermaskinen.
 - c) Den datamengden som utveksles mellom to datamaskiner som kommuniserer via nettverket.
 - d) **Ingen av alternativene er riktig.**
- 3) **Hvilken av disse lagringsenhetene er ikke en sekundærlagrings-enhet?**
 - a) En harddisk.
 - b) **En datamaskins hurtigbuffer (cache).**
 - c) En minnepinne.
 - d) Alle alternativene er sekundærlagringsenheter.

- 4) **Hva er en protokoll i nettverkssammenheng?**
- a) **Et sett kommunikasjonsregler for utveksling av data.**
 - b) En avtale mellom nettverkseier og en bedrift som bruker nettet.
 - c) Et register der all nettverkstrafikk blir lagret i henhold til kravene fra myndighetene.
 - d) Ingen av alternativene er riktig.
- 5) **Hvilket mål brukes vi vanligvis på overføringskapasitet i nettverk?**
- a) **Bits pr sekund (bps).**
 - b) Gigabyte.
 - c) Båndbredde.
 - d) Ingen av alternativene er riktig.
- 6) **Hva definerer et klient/tjener ("client/server") forhold?**
- a) Klienter tilbyr data og tjenester til tjenere.
 - b) Klienter og tjenere tilbyr data og tjenester til hverandre.
 - c) **Tjenere tilbyr data og tjenester til klienter.**
 - d) Ingen av alternativene er riktig.
- 7) **Hva definerer et "peer-to-peer" nettverk?**
- a) En er sjef, de andre er slaver.
 - b) En er slave, de andre er sjefer.
 - c) **Alle er likeverdige.**
 - d) Ingen av alternativene er riktig.
- 8) **Hvordan kan en GPS bestemme en posisjon?**
- a) En GPS beregner sin posisjon ved å lokalisere nærmeste mobile basestasjon.
 - b) **En GPS beregner sin posisjon ved å bruke lokasjonen til flere satellitter.**
 - c) En GPS beregner sin posisjon ved å bruke lokasjon til kun en satellitt.
 - d) Ingen av alternativene er riktig.
- 9) **Hva er Wi-Fi?**
- a) **Et sett av standarder for trådløs dataoverføring.**
 - b) En kvalitetsbetegnelse for trådløse nett.
 - c) Et mål på kvaliteten på en bredbåndabonnentslinje inn til huset.
 - d) Ingen av alternativene er riktig.
- 10) **En device driver er:**
- a) en spesialdatamaskin for kjøretøy.
 - b) **spesialisert programvare for input/output, slik at utstyr kan kommunisere med resten av systemet.**
 - c) enheten som holder rede på neste instruksjon som skal utføres av en prosessor.
 - d) Ingen av alternativene er riktig.
- 11) **Et maskinspråk (machine language) er:**
- a) et programmeringsspråk som oversettes av en kompilator (oversetter) til kjørbare kode.
 - b) **et binær-type programmeringsspråk bygd inn i prosessoren som datamaskinen kan kjøre direkte.**
 - c) er programmeringsspråk som er felles for alle datamaskiner slik at de kan kommunisere.
 - d) Ingen av alternativene er riktig.

- 12) **Ordstørrelse (word size) for en prosessor er:**
- a) antall ord i en tekst som kan sammenlignes i et søk.
 - b) antall bokstaver som kan behandles i en tekststreng.
 - c) **antall bit en prosessor kan prosessere på en gang.**
 - d) Ingen av alternativene er riktig.
- 13) **Ytelse for superdatamaskiner måles i:**
- a) **FLOPS.**
 - b) Gigabytes.
 - c) Antall prosessorkjerner.
 - d) Ingen av alternativene er riktig.
- 14) **Systemklokka i en datamaskin:**
- a) fordeler tiden som brukes på ulike programmer.
 - b) **bestemmer hvor raskt operasjoner i en mikroprosessor utføres.**
 - c) sørger for at dato og tid alltid er riktig satt.
 - d) Ingen av alternativene er riktig.
- 15) **Hovedformålet med forstudiefasen (fase 1) i utvikling av informasjonssystemer er:**
- a) Dokumentere krav til systemet.
 - b) Programmere systemet.
 - c) **Gjennomføre en forberedende analyse.**
 - d) Ingen av alternativene er riktig.
- 16) **Hva vil det si å vedlikeholde et informasjonssystem?**
- a) Rette opp eksisterende feil i systemet.
 - b) Utføre endringer i systemet basert på nye betingelser.
 - c) Oppdatere dokumentasjon.
 - d) **Alle alternativene er riktig.**
- 17) **Hva gjør en enhetstest?**
- a) Tester at ulike deler av systemet fungerer sammen på korrekt måte.
 - b) Tester at selve datamaskinen (maskinvaren) fungerer.
 - c) **Tester individuelle deler av programvaren.**
 - d) Ingen av alternativene er riktig.
- 18) **Hva er en algoritme?**
- a) Krav som stilles til et dataprogram.
 - b) En test for å finne feil i et dataprogram.
 - c) **En presis beskrivelse av operasjoner som skal utføres for å løse et problem.**
 - d) Ingen av alternativene er riktig.
- 19) **Hva er et flytskjema?**
- a) **Grafisk representasjon av en algoritme.**
 - b) Et skjema for å fylle inn informasjon på en webside.
 - c) Et skjema som dokumenterer sikkerhet i et databasesystem.
 - d) Ingen av alternativene er riktig.
- 20) **Hva står ACID for innen databaser?**
- a) Appropriate, Cynical, Isolation, Development.
 - b) Appropriate, Collaborative, Irrelevant, Driver.
 - c) **Atomicity, Consistency, Isolation, Durability.**
 - d) Ingen av alternativene er riktig.

Oppgave 2 – Grunnleggende programmering (25%)

Oppgave 2 a) (3 %)

```
function eLength = edgeLength(ax, ay, bx, by)
% computes the length of the edge between points (ax, ay) and
(bx, by)

    eLength = sqrt( (ax-bx)^2 + (ay-by)^2 );

end % function
```

Oppgave 2 b) (8 %)

```
function cf = circumference(polygon)
% computes the circumference of the polygon

    n = length(polygon);

    if (n < 2) || (rem(n,2) == 1)
        % not a polygon
        cf = -1;
    else
        cf = 0;

        % all but the last edge
        for i = 1:2:n-3
            cf = cf + edgeLength(polygon(i), polygon(i+1), ...
                polygon(i+2), polygon(i+3));
        end

        % adds the last edge
        cf = cf + edgeLength(polygon(n-1), polygon(n), ...
            polygon(1), polygon(2));
    end % if

end % function
```

Oppgave 2 c) (6 %)

```
function er = enclosingRectangle(polygon)
% computes the enclosing rectangle of a polygon

% selects x and y values
xValues = polygon(1:2:end);
yValues = polygon(2:2:end);

% the minimum vertex
er(1) = min(xValues);
er(2) = min(yValues);

% the maximum vertex
er(3) = max(xValues);
er(4) = max(yValues);

end % function
```

Oppgaven kan også løses ved å bruke løkker.

Oppgave 2 d) (8 %)

```
function polygon = readPolygonFile(fileName)
% reads polygon vertices from file

% opens the file
fh = fopen(fileName, 'r');
if fh == -1
    % unable to open file
    polygon = -1;
else
    % index of next number
    i = 1;
    % reads all file lines with points
    while ~feof(fh)
        fileLine = fgetl(fh);
        polygon(i:i+1) = str2num(fileLine);
        i = i + 2;
    end % while

    % closes the file
    status = fclose(fh);
    if status ~= 0
        % unable to close file
        polygon = -1;
    end % if

end % if

end % function
```

Oppgave 3 – Kodeforståelse (10%)

Oppgave 3 a) (4 %)

Det returneres: 2

Oppgave 3 b) (3 %)

Funksjonen returnerer antall like elementer, i samme posisjon, i de n første posisjonene i a og b.

Oppgave 3 c) (3 %)

Det returneres: 8

Oppgave 4 – Programmering (40 %)

Oppgave 4 a) (5 %)

```
function pos = check_highscore(points, scores)

if points > scores(10,2)
    pos = 10;
    while ( pos > 1 ) && ( points > scores(pos-1,2) )
        pos = pos - 1;
    end % while
else
    pos = -1;
end % if

end % function
```

Oppgave 4 b) (5 %)

```
function print_highscores(scores)

    for pos = 1:10
        fprintf('%2d. %d%5d\n', pos, scores(pos,:));
    end %for

end % function
```

Utskriftslinjen kan alternativt skrives som:

```
fprintf('%2d. %d%5d\n', pos, scores(pos,1), scores(pos,2));
```

Oppgave 4 c) (10 %)

```
function scores = add_highscore(points, number, scores)

    newpos = check_highscore(points, scores);

    if newpos ~= -1
        for pos = 10:-1:newpos+1
            scores(pos,:) = scores(pos-1,:);
        end % for
        scores(newpos,1) = number;
        scores(newpos,2) = points;
    end % if

end % function
```

Oppgave 4 d) (10 %)

```
function number = most_highscores(scores)

    number = scores(1,1);
    maxns = count_scores(number, scores);

    for pos = 2:10
        ns = count_scores( scores(pos,1), scores );
        if ns > maxns
            maxns = ns;
            number = scores(pos,1);
        end % if
    end % for

    if maxns == 1
        number = 0;
    end % if

end % function

function n = count_scores(number, scores)
    n = sum( (scores(:,1) == number) );
end % function
```

Oppgave 4 e) (10 %)

```
function scores = new_highscorelist

numbers = [11000000, 44000000, 22000000, 90909090,...
           73500000, 73000000, 22220000, 54000000,...
           30303030, 40404040];

pos = 1;

for points = 100:-10:10
    index = randi( length(numbers) );
    scores(pos, 1) = numbers(index);
    scores(pos, 2) = points;
    numbers(index) = [];
    pos = pos + 1;
end % for

end % function
```