

NTNU
Norges teknisk-naturvitenskapelige
universitet

Fakultetet for informasjonsteknologi,
matematikk og elektroteknikk

Institutt for datateknikk og
informasjonsvitenskap

BOKMÅL



Sensurfrist: 3. september 2012

Kontinuasjoneksamen i TDT4105
Informasjonsteknologi, grunnkurs
Mandag 13. august 2012
9.00 – 13.00

Faglig kontakt under eksamen:
Roger Midtstraum (995 72 420)

Hjelpemidler: C

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

Sensur:

Resultater gjøres kjent på studweb.ntnu.no.

Oppgavesettet inneholder 4 oppgaver. Det er angitt i prosent hvor mye hver oppgave og hver deloppgave teller ved sensur. Les igjennom hele oppgavesettet før du begynner å lage løsning. Disponer tiden godt! Gjør rimelige antagelser der du mener oppgaveteksten er ufullstendig, skriv kort hva du antar.

Svar kort og klart, og skriv tydelig. Er svaret uklart eller lenger enn nødvendig trekker dette ned.

Lykke til!

Innhold:

- Oppgave 1: Flervalgsoppgave (30 %)
- Oppgave 2: Kodeforståelse (10 %)
- Oppgave 3: Grunnleggende programmering (20 %)
- Oppgave 4: Mer programmering (40 %)
- Appendiks: Nyttige funksjoner
- Svarark til Flervalgsoppgave

Oppgave 1: Flervalgsoppgave (30 %)

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 kalles typen datamaskin som typisk brukes til tyngre oppgaver som værberegninger?**
 - a) Arbeidsstasjon.
 - b) Mikrodatamaskin.
 - c) Stormaskin.
 - d) Superdatamaskin.

- 2) Hvilket av disse alternativene beskriver IKKE en mikrokontroller?**
 - a) Kalles også innebygd datamaskin.
 - b) Brukes som en del av diverse utstyr og kjøretøyer.
 - c) Benytter alltid store sekundærlager.
 - d) Finnes i mikrobølgeovn, tastatur, klokker, etc.

- 3) Hvilket av disse alternativene beskriver IKKE programvare (software)?**
 - a) En CD-plate.
 - b) Operativsystemet.
 - c) Utstyrskjører (device drivers).
 - d) Hjelpesprogrammer (utility programs).

- 4) Hvilket alternativ beskriver sekundærminne?**
 - a) Harddisk.
 - b) CD/DVD.
 - c) Minnepenn.
 - d) Alle alternativene er riktig.

- 5) Hvilket alternativ beskriver IKKE en utenhet (output) i en datamaskin?**
 - a) Monitor.
 - b) CPU.
 - c) Høytaler.
 - d) Printer.

- 6) Hvor mange bytes er 1 kilobyte?**
 - a) 1000 bytes.
 - b) 1024 bytes.
 - c) 1048 bytes.
 - d) Ingen av alternativene er riktig.

- 7) Hva er korrekt angående systemprogramvaren i en datamaskin?**
 - a) Gjør det mulig for applikasjonsprogramvare å kommunisere med datamaskin.
 - b) Hjelper datamaskinen til å håndtere sine egne interne og eksterne ressurser.
 - c) Inkluderer utstyrskjører (device drivers) og hjelpesprogrammer (utility programs).
 - d) Alle alternativene er riktige.

- 8) Hva legges i begrepet "booting"?**
- Prosesen å laste inn operativsystemet i minne til en datamaskin.
 - Prosesen å laste inn applikasjonsprogramvare i minne til en datamaskin.
 - Prosesen å skifte mellom to programmer i et operativsystem.
 - Ingen av alternativene er riktig.
- 9) Hvilket alternativ stemmer IKKE angående beskrivelser av et operativsystem?**
- Håndterer grunnleggende dataoperasjoner på et lavt nivå.
 - De fleste applikasjonsprogrammer er skrevet for å kjøre på et spesifikt operativsystem.
 - Man kan starte programmer direkte utenom operativsystemet på de fleste universelle datamaskiner (PCer).
 - Operativsystemet tar seg av oppgaver som oppstart, prosessorhåndtering, filhåndtering, oppgavehåndtering og sikkerhetshåndtering.
- 10) Hva står forkortelsen "BIOS" for?**
- Binary Intermediate Operating System.
 - Basic Input/Output System.
 - Boot In Open Software.
 - Ingen av alternativene er riktig.
- 11) Hvordan kan datamaskin med en prosessor (CPU) som kun kan kjøre en instruksjon av gangen kjøre flere programmer samtidig (multi-tasking)?**
- Det er ikke mulig.
 - Hvert program får tildelt litt prosesseringstid.
 - Datamaskinen bruker andre enheter som grafikkort til å utføre programmer.
 - Ingen av alternativene er riktig.
- 12) Hva er oppgaven til utstyrsdrivere (device drivers)?**
- Forenkle kommunikasjon mellom datamaskinen og eksternt utstyr.
 - Bestemme hvilke programmer som skal kjøres i prosessoren (CPUen).
 - Sørge for at alle eksterne enheter kan tilkobles ved hjelp av USB.
 - Ingen av alternativene er riktig.
- 13) Hva betyr begrepet båndbredde (bandwidth)?**
- Beskriver hvilke radiofrekvenser et trådløstnettverk kommuniserer over.
 - Beskriver fysisk bredde på optisk fiber eller nettverkskabler.
 - Uttrykk for hvilken type data som kan sendes over en kommunikasjonskanal.
 - Uttrykk for hvor mye data per tidsenhet som kan sendes over en kommunikasjonskanal.
- 14) Hvilken teknologi har potensielt størst overføringskapasitet?**
- ADSL.
 - Optisk fiber.
 - Koaksialkabel.
 - Wi-Fi.
- 15) Hva er en protokoll når man snakker om nettverk?**
- Et sett med regler datamaskiner følger for å overføre data elektronisk.
 - Et program som sjekker om data kan overføres mellom to datamaskiner.
 - En standard for å representere websider.
 - Ingen av alternativene er riktig.

- 16) Omtrent hvor mange unike adresser tilbyr IPv4 (Internet Protocol ver. 4)?**
- a) 4,29 millioner.
 - b) 429 millioner.
 - c) 4,29 milliarder.
 - d) 429 milliarder.
- 17) Hva består den første delen av en Uniform Resource Locator (URL) av?**
- a) Domenenavn.
 - b) Protokoll.
 - c) Type webdokument (f.eks. HTML, php, asp, jsp).
 - d) Ingen av alternativene er riktig.
- 18) Hva står HTML for?**
- a) Horizontal Text Modelling Language.
 - b) Hypertext Modelling Language.
 - c) Hypertext Markup Language.
 - d) Ingen av alternativene er riktig.
- 19) Hva ligger i begrepet ”spoofing”?**
- a) Bombardere en nettside med stor datatrafikk.
 - b) Opprette nettsider som utgir seg for å være noe annet enn de er.
 - c) Bryte seg inn i et datasystem.
 - d) Sende epost med falsk avsender.
- 20) Hva er et hovedkort (motherboard)?**
- a) Bunnplata i et PC kabinett.
 - b) Sekundærminnet i en PC.
 - c) Et kretskort i en datamaskin der enheter som CPU, RAM, og andre enheter kobles sammen.
 - d) Et minnekort for å lagre ultralydbilder.
- 21) Hva gjør en enhetstest?**
- a) Tester individuelle deler av programvaren.
 - b) Tester at selve datamaskinen (maskinvaren) fungerer.
 - c) Tester at ulike deler av systemet fungerer sammen på en korrekt måte.
 - d) Ingen av alternativene er riktig.
- 22) Hva definerer et ”peer-to-peer” nettverk?**
- a) En er slave, de andre er sjefer.
 - b) En er sjef, de andre er salver.
 - c) Alle er likeverdige.
 - d) Ingen av alternativene er riktig.
- 23) På nettsiden www.ntnu.no/student finner du en lenke til følgende dokument ”info.html”. Hvilken type lenke er dette?**
- a) Absolutt hyperlenke.
 - b) Relativ hyperlenke.
 - c) Modifiserende hyperlenke.
 - d) Ingen av alternativene er riktig.

24) Hva er et flytskjema innen programmering?

- a) En grafisk representasjon av en algoritme.
- b) En presis beskrivelse av en endelig serie operasjoner som skal utføres for å løse et problem.
- c) En tekstlig beskrivelse som beskriver flyten i et program.
- d) Ingen av alternativene er riktig.

25) Hva kalles den siste fasen i livssyklusen til systemutvikling av informasjonssystemer?

- a) Design.
- b) Utvikling.
- c) Vedlikehold.
- d) Ingen av alternativene er riktig.

Oppgave 2 – Kodeforståelse (10 %)**Oppgave 2 a) (5 %)**

Gitt funksjonen `unknown` vist under.

```
function [a b c] = unknown(a, b, c)

    if (b >= a) && (a > 10)
        b = a;
    elseif (a <=c) && (b >= a)
        if (c ~= a)
            c = a;
        elseif (b > c)
            a = b;
        else
            b = a;
        end
    else
        b = c;
    end
end
```

Anta at $x=4$, $y=6$ og $z=4$.

Hva blir verdien til x , y og z etter at vi har utført setningen

```
[x y z] = unknown(x, y, z)?
```

Oppgave 2 b) (5 %)

Gitt funksjonen `unknown2` vist under.

```
function table = unknown2(table)

    a = table(1);
    for i = 1:length(table)-1
        table(i) = table(i+1);
    end

    table(length(table)) = a;
end
```

Anta at $B = [1\ 3\ 6\ 8\ 12]$.

Hva blir verdien til C etter at vi har utført setningen

```
C = unknown2(B)?
```

Oppgave 3 – Grunnleggende programmering (20 %)

Skøyteløp arrangeres på en rundbane som er 400 m. Et løp på 1500 m består for eksempel av 300 m pluss tre hele runder. For hver passering av målstreken tar man en passeringstid, den siste målpaseringen er resultatet i løpet.

Det er hensiktsmessig å bruke funksjonene du lager utover i oppgaven. Du kan bruke funksjoner fra andre deloppgaver selv om du ikke har klart å løse deloppgaven der du skal lage funksjonen. Du kan forutsette at alle inndata er riktige.

Oppgave 3 a) (5 %)

Lag funksjonen `mshd2s(minutter, sekunder, hundredeler)` som konverterer tid i minutter, sekunder og hundredeler til sekunder.

For eksempel skal `mshd2s(2,10,20)` returnere 130.20.

Oppgave 3 b) (5 %)

Rundetiden er tiden mellom to målpaseringer. Lag funksjonen `rundeTid(startTid, sluttTid)` som regner ut rundetiden (i sekunder med desimaler) når `startTid` og `sluttTid` er vektorer med minutter, sekunder og hundredeler.

For eksempel skal `rundeTid([0 45 20], [1 14 55])` returnere 29.35.

Oppgave 3 c) (10 %)

Lag funksjonen `alleRundeTider(passeringsTider)` som returnerer en vektor med alle rundetider (sekunder med desimaler) for et skøyteløp. Parameteren `passeringsTider` er en matrise med alle passeringstider, der hver rad har data for en målpasering på formatet minutter, sekunder og hundredeler.

Et løp over 1500 m kan for eksempel ha `passeringsTider` med verdiene `[0 20 0; 0 50 10; 1 21 21; 1 53 33]`. Dersom `alleRundeTider` kalles opp med denne matrisen som innparameter, skal funksjonen returnere vektoren `[30.10 31.11 32.12]`.

Oppgave 4 – Mer programmering (40 %)

Vi har data fra en pulsklokke som er en vektor med hjerterefrekvensen (antall pulsslag pr sekund) for hver sekund av en treningstur. Data for en treningstur på 20 sekunder kan for eksempel være:

```
[110 118 125 127 127 130 129 131 132 134 134 135 145 157 165 172 173 178 179 178]
```

Det kan være hensiktsmessig å bruke funksjonene du lager utover i oppgaven. Du kan bruke funksjoner fra andre deloppgaver selv om du ikke har klart å løse deloppgaven der du skal lage funksjonen. Du kan forutsette at alle inndata er riktige.

Oppgave 4 a) (5 %)

Lag en funksjon `pulsStatistikk(pulsData)` som returnerer gjennomsnittspuls, laveste puls og høyeste puls ut fra verdiene i innparameteren. Denne deloppgaven skal løses uten bruk av innebygde funksjoner som `min`, `max` og `mean`.

Dersom `pulsStatistikk` kalles opp med `pulsData` som vist i innledningen til oppgaven, skal det returneres 143.95, 110 og 179.

Oppgave 4 b) (5 %)

Olympiatoppen definerer 5 treningssoner med utgangspunkt i en persons makspuls. Treningssone 1 er belastning (hjerterefrekvens) fra og med 60 % til 72,5 % av makspuls. Treningssone 2 er puls fra og med 72,5 % til 82,5 %, treningssone 3 er puls fra og med 82,5 % til 87,5 %, treningssone 4 er puls fra og med 87,5 % til 92,5 %, og treningssone 5 er puls fra og med 92,5 % av makspuls og høyere. Aktivitet med puls under 60 % av makspuls regnes ikke med i noen treningssone.

Lag funksjonen `pulsSoneGrenser(maksPuls)` som returnerer en vektor med sonegrensene i pulsslag for en person med oppgitt makspuls.

For eksempel skal `pulsSoneGrenser(188)` returnere vektoren [112.8000 136.3000 155.1000 164.5000 173.9000].

Oppgave 4 c) (15 %)

Lag en funksjon `pulsSoner(maksPuls, pulsData)`. Funksjonen skal beregne hvor stor andel i prosent av den totale treningstiden en person med oppgitt `maksPuls` og med `pulsData` som oppgitt i parameteren `pulsData`, har vært i hver av de fem treningssonene. Prosenttall for de fem treningssonene skal returneres i en vektor med fem elementer.

Dersom `pulsSoner` kalles opp med makspuls på 188 og `pulsData` som vist i innledningen av oppgaven, skal resultatet bli vektoren [55 5 5 15 15]. I dette tilfellet vil utøveren for eksempel ha vært 55 % av tiden i treningssone 1.

Oppgave 4 d) (15 %)

Lag en funksjon `lengstePulsOkning(pulsData)` som finner det lengste tidsintervallet i `pulsData` der pulsen ikke går ned fra et sekund til det neste. Funksjonen skal returnere lengden (i sekunder) av dette intervallet og starttiden for intervallet. Dersom det finnes flere intervaller med samme lengde, skal funksjonen returnere det intervallet som kommer først i `pulsData`.

Dersom `lengstePulsOkning` kalles opp med `pulsData` som vist i innledningen av oppgaven, skal funksjonen returnere et intervall på 13 sekunder som starter det 7 sekundet.

Appendiks: Nyttige funksjoner

FIX Round towards zero.

FIX(X) rounds the elements of X to the nearest integers towards zero.

FLOOR Round towards minus infinity.

FLOOR(X) rounds the elements of X to the nearest integers towards minus infinity.

FCLOSE Close file.

ST = FCLOSE(FID) closes the file associated with file identifier FID, which is an integer value obtained from an earlier call to FOPEN. FCLOSE returns 0 if successful or -1 if not.

FEOF Test for end-of-file.

ST = FEOF(FID) returns 1 if the end-of-file indicator for the file with file identifier FID has been set, and 0 otherwise. The end-of-file indicator is set when a read operation on the file associated with the FID attempts to read past the end of the file.

FGETL Read line from file, discard newline character.

TLINE = FGETL(FID) returns the next line of a file associated with file identifier FID as a MATLAB string. The line terminator is NOT included. Use FGETS to get the next line with the line terminator INCLUDED. If just an end-of-file is encountered, -1 is returned.

FOPEN Open file.

FID = FOPEN(FILENAME,PERMISSION) opens the file FILENAME in the mode specified by PERMISSION:

- 'r' open file for reading
- 'w' open file for writing; discard existing contents
- 'a' open or create file for writing; append data to end of file
- 'r+' open (do not create) file for reading and writing
- 'w+' open or create file for reading and writing; discard existing contents
- 'a+' open or create file for reading and writing; append data to end of file

FPRINTF Write formatted data to file.

COUNT = FPRINTF(FID,FORMAT,A,...) formats the data in the real part of array A (and in any additional array arguments), under control of the specified FORMAT string, and writes it to the file associated with file identifier FID. COUNT is the number of bytes successfully written. FID is an integer file identifier obtained from FOPEN. It can also be 1 for standard output (the screen) or 2 for standard error. If FID is omitted, output goes to the screen.

FORMAT is a string containing ordinary characters and/or C language conversion specifications. Conversion specifications involve the character %, optional flags, optional width and precision fields, optional subtype specifier, and conversion characters d, i, o, u, x, X, f, e, E, g, G, c, and s.

The special formats \n,\r,\t,\b,\f can be used to produce linefeed, carriage return, tab, backspace, and formfeed characters respectively. Use \\ to produce a backslash character and %% to produce the percent character.

LENGTH Length of vector.

LENGTH(X) returns the length of vector X. It is equivalent to MAX(SIZE(X)) for non-empty arrays and 0 for empty ones.

MOD Modulus after division.

MOD(x,y) is $x - n \cdot y$ where $n = \text{floor}(x./y)$ if $y \neq 0$.

RAND Uniformly distributed pseudorandom numbers.

R = RAND(N) returns an N-by-N matrix containing pseudorandom values drawn from the standard uniform distribution on the open interval(0,1). RAND(M,N) or RAND([M,N]) returns an M-by-N matrix.

RANDI Pseudorandom integers from a uniform discrete distribution.

R = RANDI(IMAX,N) returns an N-by-N matrix containing pseudorandom integer values drawn from the discrete uniform distribution on 1:IMAX. RANDI(IMAX,M,N) or RANDI(IMAX,[M,N]) returns an M-by-N matrix.

REM Remainder after division.

REM(x,y) is $x - n \cdot y$ where $n = \text{fix}(x./y)$ if $y \neq 0$.

SIZE Size of array.

D = SIZE(X), for M-by-N matrix X, returns the two-element row vector D = [M,N] containing the number of rows and columns in the matrix.

SQRT Square root.

SQRT(X) is the square root of the elements of X.

STR2NUM Convert string matrix to numeric array.

X = STR2NUM(S) converts a character array representation of a matrix of numbers to a numeric matrix. For example,

S = ['1 2' ; '3 4'] str2num(S) => [1 2;3 4]

SUM Sum of elements.

S = SUM(X) is the sum of the elements of the vector X. If X is a matrix, S is a row vector with the sum over each column.

Svarskjema flervalgsoppgave

Kandidatnummer: _____ Program: _____

Fagkode: _____ Dato: _____

Antall sider: _____ Side: _____

| <i>Oppgavenr</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 | | | | |
| 1.21 | | | | |
| 1.22 | | | | |
| 1.23 | | | | |
| 1.24 | | | | |
| 1.25 | | | | |

This page is intentionally left blank

Svarskjema flervalgsoppgave

Kandidatnummer: _____ Program: _____

Fagkode: _____ Dato: _____

Antall sider: _____ Side: _____

| <i>Oppgavenr</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 | | | | |
| 1.21 | | | | |
| 1.22 | | | | |
| 1.23 | | | | |
| 1.24 | | | | |
| 1.25 | | | | |