

## Eksamensoppgave i TDT4105 IT–grunnkurs, med Matlab

**Faglig kontakt under eksamen:**Rune Sætre  
Anders ChristensenMobil: **452 18103**  
Mobil: **918 97181****Eksamensdato:****2016-12-06****Eksamenstid (fra-til):****09:00 – 13:00****Hjelpemiddelkode/Tillatte hjelpemidler:****D – Bare bestemt, enkel kalkulator**

### Annen informasjon:

Opgavesettet 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 løse oppgavene. 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.

HUSK å rive løs og levere et flervalgsoppgave svarark!

### Innhold:

- Oppgave 1: Flervalgsoppgave (25%)
- Oppgave 2: Kodeforståelse (20%)
- Oppgave 3: Programmering Valg (25%)
- Oppgave 4: Programmering Penger (30%)
- Appendix: Potensielt nyttige funksjoner
- Svarskjema til flervalgsoppgave (2 eksemplarer)

**Målform/språk:****Bokmål****Antall sider:****17 (inkludert Forside, Appendiks og 2x Svarark)****Informasjon om trykking av eksamensoppgave****Originalen er:****1-sidig**  **2-sidig** **sort/hvit**  **farger** **skal ha flervalgskjema** **Kontrollert av:**

15. des. 2016

Dato

Sign

## 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 forventet klokkehastighet på en moderne datamaskin?
  - a. Litt over 1 kHz
  - b. Litt over 1 MHz
  - c. Litt over 1 GHz
  - d. Litt over 1 THz
2. Hva skiller en Solid-State Drive (SSD) fra andre Hard-Drives (HD)?
  - a. Den inneholder mer solide magnet-plater.
  - b. Den er basert på en gammeldags produksjonsteknikk.
  - c. Den har ingen bevegelige deler.
  - d. Den spinner mye raskere.
3. Hva er den vanligste bruken av Cyan-Magenta-Yellow-Key (CMYK)?
  - a. Blande farger i nøkkelmagasinet
  - b. Blande farger på en skjerm
  - c. Blande farger på en prosjektør
  - d. Blande farger på en utskrift
4. Hvor stor er de minste moderne transistorene i IC-kretser?
  - a. Ca. 10 mm
  - b. Ca. 10 mikrometer
  - c. Ca. 10 nanometer
  - d. Ca. 10 picometer
5. Hvilket steg i «fetch-execute cycle» flytter verdiene fra Minnet til ALU?
  - a. IF
  - b. ID
  - c. DF
  - d. EX
6. Hva er det største tallet som kan representeres med et 64 bits signed integer som «two's complement»?
  - a.  $2^{64}-1$
  - b.  $2^{63}-1$
  - c.  $2^{65}$
  - d.  $2^{64}$
7. Hva er tallverdien i desimalsystemet av det heksadesimale tallet 23F?
  - a. 575
  - b. 792
  - c. 1048
  - d. 312
8. I RGB-systemet med 24-bits fargekoder, hvilken farge er FFFF00
  - a. rødt
  - b. grønt
  - c. grått
  - d. gult

9. Hva er representasjon i binærtallsystemet av det desimale tallet 1234?
  - a. 0100 1101 0010
  - b. 0010 0101 1011
  - c. 0101 1011 0111
  - d. 1101 0011 0100
10. Hvordan påvirker sampleraten kvaliteten i en analog-til-digital-konvertering av lyd?
  - a. den påvirker ikke kvaliteten
  - b. høyere samplerate gjør at man får med høyere frekvenser
  - c. høyere samplerate gjør at man får med sterkere lyder
  - d. høyere samplerate gjør at man produserer mindre data
11. Hva er pakkeswitching?
  - a. En metode en maskin bruker for å motta meldinger
  - b. En protokoll en maskin bruker for å kontakte en annen maskin på nettet
  - c. En måte å dele opp meldinger i mindre pakker som kan sendes over nettet
  - d. En avleggs metode ARPANet brukte inntil 1970 til å sende meldinger på nettet
12. I hvilket av lagene i OSI-modellen finner vi IP-protokollen?
  - a. Application
  - b. Transport
  - c. Network
  - d. Link
13. Hvilken av disse feildeteksjonsmetodene er best til å oppdage "burst"-feil?
  - a. Paritet
  - b. Sjekksum
  - c. CRC
  - d. RAC
14. Hva menes med "den digitale dividende"?
  - a. betegnelse på forskjellen mellom en digital og en analog ressurs
  - b. betegnelsen på ledige (ikke-tildelte) frekvensressurser i det elektromagnetiske spektrum
  - c. forskjellen mellom en digitalt kompetent og en ikke digitalt kompetent person
  - d. uttrykk for en digital divisjonsoperasjon i en CPU
15. Hvordan kan vi forsikre oss om at avsender er den han/hun sier han/hun er?
  - a. Avsenderen har kryptert meldingen med sin private nøkkel
  - b. Avsenderen har kryptert meldingen med sin offentlige nøkkel
  - c. Avsenderen har brukt sjekksum og paritet
  - d. Mottagerens e-post program kan sammenligne mottatt e-post med den som ligger i avsenderens ut-post

Følgende kode viser en Matlab-implementasjon av en sorteringsalgoritme kjent som bubblesortering. Innparameteren `li` er ei liste, og funksjonen skal returnere samme liste sortert. Følgende tre spørsmål (16,17,18) angår algoritmen vist i figuren under.

```

function li = bubblesort(li)
swapped = true;
while (swapped)
    swapped = false;
    for i = 2: length(li)
        if (li(i-1) > li(i))
            temp = li(i);
            li(i) = li(i-1);
            li(i-1) = temp;
            swapped = true;
        end
    end
end
end %function

```

16. Vil algoritmen fungere for å sortere elementene i lista li?
- Ja, den vil sortere stigende.
  - Ja, den vil sortere synkende.
  - Den vil bare fungere for lister av heltall, ikke flyttall.
  - Nei, den vil ikke fungere.
17. Forutsatt at algoritmen fungerer, hva er dens tidskompleksitet (worst case) hvis vi betegner lengda av lista som n?
- $\Theta(n)$
  - $\Theta(n \log n)$
  - $\Theta(n^2)$
  - $\Theta(n^3)$
18. Best case for denne algoritmen er hvis lista tilfeldigvis er riktig sortert allerede, hva er tidskompleksiteten da?
- $\Theta(\log n)$
  - $\Theta(n)$
  - $\Theta(n \log n)$
  - $\Theta(n^2)$
19. En funksjon utfører flyttallsmultiplikasjoner på tall som den får inn som argument i form av ei liste. Hvis antall element i lista er n, kan antall flyttallsmultiplikasjoner som utføres uttrykkes som  $f(n) = 2n^3 - n^2 - 3n + 10$ . Hva blir tidskompleksiteten for denne funksjonen?
- $\Theta(n)$
  - $\Theta(n \log n)$
  - $\Theta(n^2)$
  - $\Theta(n^3)$
20. Hva er betingelsen for å kunne bruke binærsøk på ei liste av tall?
- Tallene bruker bare 0 og 1 som siffer.
  - Lista er sortert stigende.
  - Lista er sortert, enten stigende eller synkende.
  - Lista har et odde antall elementer.
21. En kjent ulempe med inkrementell utvikling er at
- Tidskompleksiteten til systemet stiger kvadratisk med antall inkremitter
  - Kostnaden til systemet stiger kvadratisk med antall inkremitter
  - Inkrementelle algoritmer kun fungerer for sorterte data
  - Strukturen til systemet har en tendens til å gradvis forringes.

22. Med gjenbruk av programvare (software reuse) menes i pensum at ...
- flere brukere benytter samme abonnement for en app for å spare penger
  - en programmodul skrevet for ett system også kan benyttes i et annet
  - gamle mikroprosessorer kan smeltes om så materialene kan gjenbrukes
  - straks vi har skrevet et program, kan vi kjøre det om og om igjen
23. Fire sentrale designaktiviteter nevnt i teoriboka er arkitekturdesign, databasedesign, systemgrensesnittdesign (system interface design) og ...
- fildesign
  - komponentdesign
  - løkkedesign
  - matrisedesign
24. Innen systemutvikling (software engineering) betyr iterativ utvikling (iterative development)
- Man benytter løkker i programkoden
  - Man benytter rekursjon i programkoden
  - At man gjør en ovenfra og ned (top-down) inndeling av systemet
  - At man leverer stadig nye versjoner av systemet, med gradvis økende funksjonalitet
25. Begrepet sort-boks testing (black box testing) innebærer at
- Man tester programmets kun ut fra dets ytre grensesnitt, ikke indre oppbygning
  - At man prøver å hacke programmet for å finne sikkerhetshull
  - At man tester programmet samtidig som det er i operasjon
  - At man tester programmets indre, uten tanke på brukergrensesnittet

## Oppgave 2: Kodeforståelse (20%)

### Oppgave 2a (5%)

Hva blir skrevet ut til skjerm når du kjører funksjonen `main` i koden vist under? (3%)

Forklar med en setning hva funksjonen `flopp` gjør (2%)

```
function main ()
M = [0 1 0; 1 0 1; 0 1 0];
M = flopp(M);
disp(M);
end %function

function mat = flopp (mat)
[r, c] = size(mat);
for i = 1: r
    for j = 1:c
        if mat (i, j)
            mat (i, j) = 0;
        else
            mat (i, j) = 1;
        end
    end
end %for
end %function
```

**Oppgave 2b (5%)**

Hva blir returnert hvis koden `compute (30,11,1337)` med kode som vist under blir kjørt? (3%)

Forklar med en setning hva funksjonen `compute` gjør (2%)

```
function t = compute (d, m, y)
M = {'Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', ...
     'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'};
O = {'st', 'nd', 'rd', 'th'};

ord = mod (mod (d, 30), 20);
if (ord > 4 || ord == 0)
    ord = 4;
end

t = sprintf ('%i%s %s %i', d, O{ord}, M{m}, y);
end %function
```

**Oppgave 2c (5%)**

Hva blir returnert hvis `[a, b] = fr('abcbcabd')` med kode som vist under blir kjørt? (3%)

Forklar med en setning hva funksjonen `fr` gjør (2%)

```
function [b, a] = fr (s)
f = zeros (1,26);
for i=1: length(s)
    if s(i)>='a' && s(i)<='z'
        f(s(i)-'a'+1) = f(s(i)-'a'+1) + 1;
    end
end
a = max(f);
b = char(find(f==max(f)) + 'a'-1);
end
```

**Oppgave 2d (5%)**

Hva blir returnert hvis `f (32145)` med kode som vist under blir kjørt? (3%)

Forklar med en setning hva funksjonen `f` gjør (2%)

```
function y = f (x)
y = 0;
while x > 0
    y = y + rem (x, 10);
    x = floor (x / 10);
end
if y >= 10
    y = f (y);
end
end
```

### Oppgave 3: Programmering Valg (25%)

I et land langt borte skal det avholdes valg. Hvert valgdistrikt telles manuelt. Du skal skrive kode for å regne sammen resultatene som rapporteres inn. Landet er delt inn i 92 valgdistrikt som hver velger én representant til parlamentet, samt at presidenten velges ved flertall blant de stemmeberettigete på tvers av alle valgdistriktene.

Du kan anta at innparameterne til funksjonene alltid inneholder lovlige verdier.

#### Oppgave 3a: Initialisering (5%)

Det er registrert fem ulike partier i landet, og navnene er definert i et cell array kalt `parties`. Skriv en funksjon `initElection` som definerer og pre-allokerer plass til en tabell for å lagre stemmetall for hvert parti i hvert valgdistrikt, og la funksjonen returnere denne tabellen initialisert til situasjonen før valget - dvs at alle har 0 stemmer.

Eksempel på bruk av funksjonen til å opprette tabellen med 92 linjer og en kolonne for hvert parti, pluss utlisting av de tre første linjene er vist her:

```
>> parties = {'Tea Party', 'Coffee Party', 'Milk Party', ...
             'House Party', 'Beach Party'};
>> election = initElection(parties);
>> election (1:3, :)
ans =
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
>>
```

#### Oppgave 3b: Oppdatering (5%)

Etterhvert som resultatene strømmer på, må de settes inn i tabellen som ble opprettet av funksjonen i oppgave a. Skriv en funksjon `updateElection` som tar tre parametere inn: tabellen med valgdata, nummeret på valgdistriktet, og en vektor med stemmetall for de deltagende partiene, i samme rekkefølge som de er listet i variabelen `parties`. Hvert valgdistrikt kan ha flere stemmelokaler, slik at det kan bli lagt inn stemmer for et distrikt flere ganger. Nye tall som kommer inn skal plusses sammen med tall som ligger der fra før. Funksjonen skal returnere en oppdatert tabell.

Eksempel på kjøring:

```
>> election = initElection(parties);
>> election = updateElection (election, 34, [123,3321,3442,23,1]);
>> election (34, :)
ans =
    123    3321    3442     23     1
>> election = updateElection (election, 34, [601,2000,3000,50,22]);
>> election (34, :)
ans =
    724    5321    6442     73     23
>>
```

**Oppgave 3c: Presidentvalget (5%)**

Skriv en funksjon `printLeadP` som tar resultatene i tabellen `election` med valgdata, og regner ut hvilket parti som totalt har fått flest stemmer, og som skriver ut at dette partiet leder.

I denne del-oppgaven kan du gjøre en forenklet antagelse at funksjonen `printLeadP` kun kalles etter at det har blitt registrert noen stemmer (dvs. ikke hele tabellen er null), og at det ikke er delt ledelse mellom flere partier, dvs. ett unikt parti har flest stemmer.

Eksempel på kjøring (NB: eksemplene her og videre utover antar at det er satt inn vesentlig mer data i tabellen enn bare det som ble gjort i eksemplet i 3b):

```
>> printLeadP (election, parties)
Milk Party is leading the election with 341433 votes
>>
```

**Oppgave 3d: Parlamentsvalget (10%)**

Siden det kun er en person som velges i hver enmannskrets, er det partiet med flest stemmer i kretsen som får valgt inn sin kandidat.

Skriv funksjonen `printResults` som lister ut gjeldende mandatfordeling i parlamentet basert på data som hittil har kommet inn for valgdistriktene. For å få full pott må tallene komme ut høyrejustert på en jevn kolonne, som vist under. Hvis et parti kun har 1 delegat hittil, skal man skrive ordet `delegate` i entall, ellers `delegates` (også for null). Det kan antas at maksimalt antall delegater er tresifret. I denne deloppgaven skal det tas høyde for at to eller flere partier kan ligge likt i teten i et valgdistrikt. Hvis ett parti er alene i teten i et valgdistrikt, skal distriktet telles for dette partiet. Hvis flere partier deler ledelsen, skal distriktet telles som `Undecided (tied)`. Hvis det ennå ikke har blitt registrert noen stemmer i distriktet, skal det derimot telles som `Undecided (no votes)`.

Eksempel på kjøring:

```
>> printResults(election, parties)
Tea Party:           3 delegates
Coffee Party:       12 delegates
Milk Party:         1 delegate
House Party:        4 delegates
Beach Party:       20 delegates
Undecided (tied):   2 delegates
Undecided (no votes): 50 delegates
>>
```



## Oppgave 4: Programmering Penger (30%)

Firmaer som setter opp kjøpskontrakter har ofte behov for å oppgi beløpet som skal betales både med tall og tekst. Store beløp blir en lang tekst og ansatte har lett for å skrive feil slik at tall og tekst ikke stemmer overens. Man ønsker derfor et program som automatisk kan oversette tall til tekst. I denne oppgaven skal tallene omformes til engelsk tekst. Du kan anta at det i hovedprogrammet allerede er deklartert to variabler `D` og `L` som du kan benytte deg av (ved å bruke `global D L` øverst i funksjonene dine) hvis du ønsker det, men det er ikke noe krav at disse variablene må brukes.

Skriv funksjonene under slik at du kan gjenbruke dem. Hvis du ikke klarer å løse en del-oppgave kan du likevel anta at funksjonen virker som oppgitt og gjenbruke den i en senere del-oppgave. Du kan anta at funksjonene alltid får inn lovlige verdier.

`D` og `L` inneholder følgende konstante verdier (skal ikke endres underveis).

```
global D L
D = {'one', 'two', 'three', 'four', 'five', ...
     'six', 'seven', 'eight', 'nine', 'ten', ...
     'eleven', 'twelve', 'thirteen', 'fourteen', 'fifteen', ...
     'sixteen', 'seventeen', 'eighteen', 'nineteen', 'twenty', ...
     '', '', '', '', '', '', '', '', '', 'thirty', ...
     '', '', '', '', '', '', '', '', '', 'forty', ...
     '', '', '', '', '', '', '', '', '', 'fifty', ...
     '', '', '', '', '', '', '', '', '', 'sixty', ...
     '', '', '', '', '', '', '', '', '', 'seventy', ...
     '', '', '', '', '', '', '', '', '', 'eighty', ...
     '', '', '', '', '', '', '', '', '', 'ninety', ...
     '', '', '', '', '', '', '', '', ''};
L = {1000000, ' million', 1000, ' thousand', 1, ''}
```

### Oppgave 4a: 2-digit (7%)

Skriv en funksjon `i2_txt` med parameter `tt` som skal være et positivt heltall med maksimalt to siffer (dvs.  $1 \leq tt \leq 99$ ) og returnere tilsvarende tall som engelsk tekst.

Eksempler på bruk:

```
>> i2_txt (5)
ans =
five
>> i2_txt (30)
ans =
thirty
>> i2_txt (68)
ans =
sixty-eight
```

**Oppgave 4b: 3-digit (7%)**

Skriv en funksjon `i3_txt` med parameter `ttt` som skal være et positivt heltall med maksimalt tre siffer (dvs.  $1 \leq \text{ttt} \leq 999$ ) og returnere samme tall som engelsk tekst. Tall med bare ett eller to siffer skal gi samme resultat som i (a).

Eksempel på bruk med tresifrede tall:

```
>> i3_txt (100)
ans =
one hundred
>> i3_txt (345)
ans =
three hundred forty-five
```

**Oppgave 4c: 9-digit (7%)**

Skriv en funksjon `i9_txt` med parameter `tall` som skal være et positivt heltall med maksimalt ni siffer (dvs.  $1 \leq \text{tall} \leq 999\,999\,999$ ) og returnere samme tall i dets tekstlige form. Tall med tre eller færre siffer skal gi resultat som vist i (a) og (b).

Eksempel på bruk med større tall:

```
>> i9_txt (12000)
ans =
twelve thousand
>> i9_txt (276900)
ans =
two hundred seventy-six thousand nine hundred
>> i9_txt (67000020)
ans =
sixty-seven million twenty
```

**Oppgave 4d: Sett inn tekst (9%)**

Skriv en funksjon `add_words` med parameter `txt` som skal være en setning (tekststreng) som potensielt inneholder ett eller flere beløp gitt som en serie siffer, og returnere samme setning med tekstlige varianter av beløpene innskutt bak tallet omsluttet av bindestreker. Hvis man ikke finner noen tall, skal setningen være uendret, og du kan anta at det kun er snakk om heltall, ikke beløp med desimaler.

Hint: `str2num` returnerer et tall større enn null bare hvis innparameteren er et «tekst-tall» større enn null.

Eksempler på bruk:

```
>> add_words ('Mr. X shall pay 9005100 dollars')
ans =
Mr. X shall pay 9005100 - nine million five thousand one hundred - dollars
>> add_words ('C owes 91 pounds to D and 55 pounds to E')
ans =
C owes 91 - ninety-one - pounds to D and 55 - fifty-five - pounds to E
>> add_words ('The evildoer is hereby fined 945000000 yen')
ans =
The evildoer is hereby fined 945000000 - nine-hundred forty-five million - yen
```

## Appendix: Potensielt 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.

find

Returns the linear indexes of non-zero elements in a matrix. `FIND([0 1 0 1 0])` returns `[2 4]`. If the first parameter has more than one row, a column vector containing the linear indexes of non-zero elements are returned. An optional second parameter set the maximum number of indexes to return.

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.

**global**

Define global variable.

`global X Y Z` defines X, Y, and Z as global in scope (scope can be functions/programs).

**input**

Read a value from the keyboard and into a variable.

`ANSWER=INPUT(STR)` prints STR as a prompt, reads a number and assigns it to ANSWER.

If character string is to be read, use the optional second parameter 's'.

**isempty**

Determine whether array is empty

This MATLAB function returns logical 1 (true) if A is an empty array and logical 0 (false) otherwise.

`TF = isempty(A)`

**length**

The 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.

**load**

Loads data from filename.

`load(filename)` loads data from filename. If filename is a MAT-file, then `load(filename)` loads variables in the MAT-File into the MATLAB® workspace. If filename is an ASCII file, then `load(filename)` creates a double-precision array containing data from the file.

**max**

finds the highest element in a vector, or the highest element in each column of a matrix.

**min**

finds the lowest element in a vector, or the lowest element in each column of a matrix.

**mod**

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

**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  $y \neq 0$ .

**round**

Rounds to nearest decimal or integer. `Y = round(X)` rounds each element of X to the nearest integer. If an element is exactly between two integers, the round function rounds away from zero to the integer with larger magnitude. `Y = round(X,N)` rounds to N digits

**size**

The 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.

**sortrows**

Sort array rows. This MATLAB function sorts the rows of A in ascending order, based on column. `B = sortrows(A)`. `B = sortrows(A, column)`

sqrt

Square root.  $\text{SQRT}(X)$  is the square root of the elements of  $X$ .

sscanf

Extracts values from a string according to a format string. Opposite of `FPRINTF`.

$A = \text{SSCANF}('12/11-2014', '%d/%d-%d')$  returns a column vector containing the values 12, 11, and 2014.

strsplit

Splits the first (string) parameter into a cell array of substrings, according to the delimiter string given as the second parameter. `STRSPLIT('one, two, three', ',')` results in `{'one', 'two', 'three'}`. Multiple alternative delimiters can be specified using a cell array as the second parameter.

strtok

separates the first token of a string from the rest of that string.

$[\text{TOKEN}, \text{REST}] = \text{STRTOK}(' \text{first second}', \text{DELIM})$  sets `TOKEN` to `'first'` and `REST` to `'second'`. The optional parameter `DELIM` contains a list of delimiter characters – where the space character is default. Any delimiter characters before the first token are ignored.

str2num

Convert string matrix to numeric array.

$X = \text{STR2NUM}(S)$  converts a character array representation of a matrix of numbers to a numeric matrix. For example,  $S = ['12'; '34']$   $\text{str2num}(S) \Rightarrow [12; 34]$ .

$S = 'abc'$   $\text{str2num}(S) \Rightarrow []$

sum

The sum of elements.  $S = \text{SUM}(X)$  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.

*Denne siden er med hensikt blank!*

## Svarskjema til 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				

*Denne siden er med hensikt blank!*



**Svarskjema flervalgsoppgave**

Kandidatnummer: \_\_\_\_\_ Program: \_\_\_\_\_

Fagkode: \_\_\_\_\_ Dato: \_\_\_\_\_

Antall sider: \_\_\_\_\_ Side: \_\_\_\_\_

<b><i>Oppgavenr</i></b>	<b><i>A</i></b>	<b><i>B</i></b>	<b><i>C</i></b>	<b><i>D</i></b>
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				