

## Eksamensoppgave i TDT4105 Informasjonsteknologi – grunnkurs

### Innhold:

- Oppgave 1: Flervalgsoppgave (25 %)
- Oppgave 2: Grunnleggende programmering (20 %)
- Oppgave 3: Kodeforståelse (15 %)
- Oppgave 4: Mer programmering (40 %)

## LØSNINGSFORSLAG

### Oppgave 1: Flervalgsoppgave (25 %)

1A, 2C, 3D, 4C, 5B,  
6B, 7C, 8B, 9C, 10A  
11C, 12D, 13D, 14C, 15A,  
16D, 17C, 18C, 19B, 20A  
21C, 22A, 23C, 24B, 25B

### Oppgave 2: Grunnleggende programmering (20 %)

#### Oppgave 2a) (4 %)

```
function readOneNumber()  
%readOneNumber Read 1 row, column and digit and  
write message to user.  
row = input( 'Rad (1-9): ' );  
column = input( 'Kolonne (1-9): ' );  
digit = input( 'Tallet (1-9): ' );  
%board(row,column) = digit %Ikke spurt om lagring 2a  
fprintf('Posisjon (%d,%d) inneholder nå %d\n', ...  
    row, column, digit );  
end
```

8								
9								
1								
2						1	3	5
3						2	4	6
4	5	6	1	2	3	9	8	7
5								
6								
7								

#### Oppgave 2b) (4 %)

```
function board = readPositionDigit( rowNr, colNr, board )  
%readPositionDigit Read one new digit, put it in the right rowNr and colNr, and  
%return the updated board.  
prompt = sprintf( 'Verdi for posisjon (%d,%d): ', rowNr, colNr );  
digit = input( prompt );  
board(rowNr, colNr) = digit;  
end
```

### Oppgave 2c (6 %):

```
function board = readValidPositionDigit(rowNr, colNr, board)
%readValidPositionDigit Keep asking the user until a value from 0-9 is
%given, put the value in position (rowNr, colNr) and return the new board.
badDigit = true;
while ( badDigit )
    prompt = sprintf ( 'Verdi for posisjon (%d, %d): ', rowNr, colNr );
    digit = input ( prompt );
    badDigit = ( length(digit) > 1 || digit < 0 || digit > 9 );
    if ( badDigit )
        disp( 'Feil! Oppgi et siffer mellom 0 og 9...' );
    end %if bad
end %while bad
board(rowNr, colNr) = digit;
end %function
```

### Oppgave 2d (6 %):

```
function board = readSudokuBoard()
%READSUDOKUBOARD Read one column of digits at the time, top to bottom,
%then left to right
board = zeros(9,9);
for col = 1:9
    for row = 1:9
        board = readValidPositionDigit( row, col, board );
    end %row
end %col
end %function
```

## Oppgave 3 – Kodeforståelse (15 %)

### Oppgave 3 a) (6 %)

1. `>> res = o3a(6) → res = 8`
2. o3a beregner Fibonacci-tall nummer n, der n er input til funksjonen.

### Oppgave 3 b) (6 %)

1. `>> res = o3b(6) → res = 720`
2. o3b beregner n! (fakultetet av n =  $n * (n-1) * (n-2) * \dots * 1$ ), der n er input til funksjonen.

### Oppgave 3 c) (3 %)

**KodeA er MYE raskere!**

(fordi den ikke lager en ny kopi av matrisen for hver av de ca. 80.000 linjene).

## Oppgave 4: Mer programmering (40 %)

### Oppgave 4a (5 %)

```
function ok = bidOk( bid, trick )
[ digit, ~ ] = strtok( bid );           % Get the bid
bidTricks = 6 + str2double( digit );    % Convert to number and add 6
ok = ( trick >= bidTricks );           % Check if enough tricks taken
end %function
```

### Oppgave 4b (10 %)

```
function utgang = utgangOk( melding, antStikk )
[meldtStikk, rest] = strtok( melding );      %Get bid and farge
[farge, ~] = strtok( rest );                %Remove ' '-spaces
meldtStikk = str2double( meldtStikk ) +6;    %Convert string number, add 6
utgang = false;                             %"Usually" false
if ( bidOk(melding,antStikk) )               %If bidOk, check "utgang"
    if ( strcmp(farge, 'grand' ) )
        trenger = 9;
    elseif ( strcmp(farge, 'hjerter' ) || strcmp(farge, 'spar' ) )
        trenger = 10;
    else
        trenger = 11;
    end %if grand, else if major-farge, else minor-farge
    utgang = ( meldtStikk >= trenger);
end %if bidOk
end %function
```

### Oppgave 4c (15 %)

```
function points = bridgePoints( bid, tricks )
%Calculate number of points, or minus-point (For opponent)
[bidTricks, trump] = strtok(bid);
bidTricks = str2double( bidTricks ) + 6;
trump = strtok( trump ); %Remove space character
if ( bidOk(bid, tricks) )
    bonus = 50;
    if ( utgangOk(bid, tricks) )
        bonus = 300;
    end %if utgangOK
    points = bonus + trickPoints( trump, tricks );
else
    points = (tricks -bidTricks) * 50;
end %if contractOK, else beit
end %function

function points = trickPoints( trump, tricks )
%Calculate number of trick points based on trump color
tricks = tricks - 6;
if ( strcmp( 'kløver', trump ) || strcmp( 'ruter', trump ) )
    points = 20*tricks;
elseif (strcmp( 'hjerter', trump ) || strcmp( 'spar', trump ) )
    points = 30*tricks;
else
    points = 10 + 30 * tricks;
end %if minor, major, NT
end %function
```

## Oppgave 4d (10 %)

```
function bridge()
results = {}; %Cell Array
while true
    lag = input( 'Lag (N/S eller Ø/V, annet for å slutte): ', 's');
    if ( strcmp( 'N/S', lag ) == 0 && strcmp( 'Ø/V', lag ) == 0 )
        break;
    end %if ikke N/S og ikke Ø/V, stop
    bid = input( 'Melding: ', 's');
    stikk = input( 'Stikk: ' );
    points = o4c_bridgePoints( bid, stikk );
    beit = 0;
    if points < 0
        beit = -points;
        points = 0;
    end %if beit
    results = [results; {lag, bid, stikk, points, beit}];
end %while forever
printTotalPoints( results );
end %function

function printTotalPoints( results )
results
nspoints = 0;
ewpoints = 0;
[rows, cols] = size( results );
for i = 1:rows
    if strcmp( 'N/S', results{i,1} )
        nspoints = nspoints + results{i,4};
        ewpoints = ewpoints + results{i,5};
    else
        nspoints = nspoints + results{i,5};
        ewpoints = ewpoints + results{i,4};
    end %if N/S bidder, else Ø/V bidder
end %for each game
fprintf( 'Total score:\n N/S %d\n Ø/V %d\n', ...
    nspoints, ewpoints );
end %function
```