Løsningsforslag

Oppgave 1 (25%)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
a	b	a	b	c	a	c	c	b	d	d	b	b	c	a	d	b	c	**b**	c

Oppgave 2 (20%)

2a) LF: Linje 2, `midten = floor((hoyIndeks+lavIndeks)/2)`
2b) LF: [9,7,5,3,2,1], snur lista
2c) LF: **101011001**, konverterer heltall til binærrepresentasjon **# Feil i orginal LF**
2d) LF: [2, 8, 4, 10, 7, 6, 3, 5, 9, 1] (tallene 1 til 10 i tilfeldig rekkefølge), randomiserer plasseringen av 10 tall i ei liste.

Oppgave 3 (25%)

Løsningsforslag Oppgave 3:

3a) 6%

```
function table = file_to_table( filename )
    fh = fopen( filename, 'r') ;
    if fh==-1
        error(['Could not open file ',filename]) ;
    end
    idx = 1 ;
    while ~feof(fh)
        line = fgetl(fh) ;
        for j=1:6
            [tmp,line] = strtok(line,',') ;
            table{idx,j} = str2num(tmp) ;
        end
        table{idx,7} = strtok(line,',') ;
        idx = idx + 1;
    end

    fclose(fh) ;
end
```

3b) 3%

```
function diff = time_diff( start, stop )
     vect = stop - start ;
     diff = vect(6) + 60*(vect(5) + 60*( vect(4) + ...
          24*( vect(3) + 30.44*( vect(2) + 12*vect(1))))) ;
end
```

3c) 5%

```
function crazy = check_min_distance( cars, diff)
    crazy = {} ;
    for i=1:size(cars,1)-1
        dtime = time_diff( [cars{i,1:6}], [cars{i+1,1:6}]) ;
        if (dtime < diff)
            crazy{end+1} = cars{i+1,7} ;
        end
    end
end
```

3d) 4%

```
function elcars = list_el_cars( cars )
    elcars = 0 ;
    for i=1:size(cars,1)
        regno = cars{i,7}(1:2) ;
        if (regno(1)=='E' && regno(2)=='K'||regno(2)=='L'||...
                regno(2)=='V'))
            elcars = elcars + 1 ;
        end
    end
end
```

3e) 5%

```
function llist = generate_licence_numbers( amount )
    letters = { 'BS', 'CV', 'EL', 'FY', 'KU', ...
                'LE', 'NB', 'PC', 'SY', 'WC'} ;
    llist = {} ;
    while amount>length(llist)
        licno=[letters{randi(length(letters),1,1)} ...
                num2str(randi(90000,1,1)+9999) ] ;
        found = false ;
        for i=1:length(llist)
            if strcmp(llist{i},licno)
                found = true ;
            end
        end
        if ~found
            llist{end+1} = licno ;
        end
    end
end
```

3f) 7%

```
function speeders = list_speeders( filename_a, ...
                                   filename_b, speed_limit, distance)
boxa = file_to_table( filename_a ) ;
boxb = file_to_table( filename_b ) ;
tdiff = distance*1000/(speed_limit*1000/3600) ;
speeders = {} ;
for i=1:size(boxb,1)
    index = 0 ;
    for j=1:size(boxa,1)
        if strcmp(boxa{j,7},boxb{i,7})
            index = j ;
        end
    end

    if index
        if time_diff( [boxa{index,1:6}], ...
                                    [boxb{i,1:6}])<tdiff
            speeders{end+1} = boxb{i,7} ;
        end
    end
end
end
```

Oppgave 4 (30%)

Løsningsforslag oppgave 4:

4a) (3%)

```
function output = formatTime( input )
    hour = floor( input/3600 ) ;
    mins = floor( mod( input, 3600 ) / 60 ) ;
    secs = mod( input, 60 ) ;
    output = sprintf( '%02d:%02d:%02d', hour, mins, secs ) ;
end
```

4b) (2%)

```
function [first, period] = valuesDecember()
    first = 3*3600 + 18*60 ;
    period = 12*3600 + 25*60 + 12 ;
end
```

4c) (5%)

```
function [lows, highs] = genTides()
    lows = [] ;
    highs = [] ;
    [first, period] = valuesDecember() ;

    event = first ;
    while event<24*60*60*31
        lows(end+1) = event ;
        event = event + period ;
    end

    event = first + period/2 ;
    while event<24*60*60*31
        highs(end+1) = event ;
        event = event + period ;
    end
end
```

4d) (3%)

```
function output = genTidesStr( input )
    for var = 1:length(input)
        day = floor(input(var)/(24*60*60)) ;
        output{var} = sprintf( '%d %s', day+1, ...
                    formatTime( mod(input(var), 24*60*60 ))) ;
    end
end
```

4e (7%)

```
function checkTides( dayInMonth )
    [lows, highs] = genTides() ;
    start = (dayInMonth*24 + 9)*3600 ;
    duration = 4*3600 ;
    idx = find(lows>=start & lows<=(start+duration)) ;
    if idx
        fprintf('low tide at %s\n', ...
                    formatTime(mod(lows(idx(1)),24*3600)));
    else
        idx = find(highs>=start & lows<=(start+duration)) ;
        if idx
            fprintf('high tide at %s\n', ...
                    formatTime(mod(highs(idx(1)),24*3600)));
        else
            fprintf('no tides\n');
        end
    end
end
```

4f (5%)

```
function listTides()
    [lows, highs] = genTides() ;
    index = 1 ;
    fprintf( 'day  first    second\n') ;
    for day=1:31
        fprintf( ' %2d', day) ;
        while (index<=length(lows) && lows(index)<day*24*60*60)
            fprintf( ' %s', formatTime(mod(lows(index),24*60*60))) ;
            index = index + 1 ;
        end
        fprintf('\n');
    end
end
```