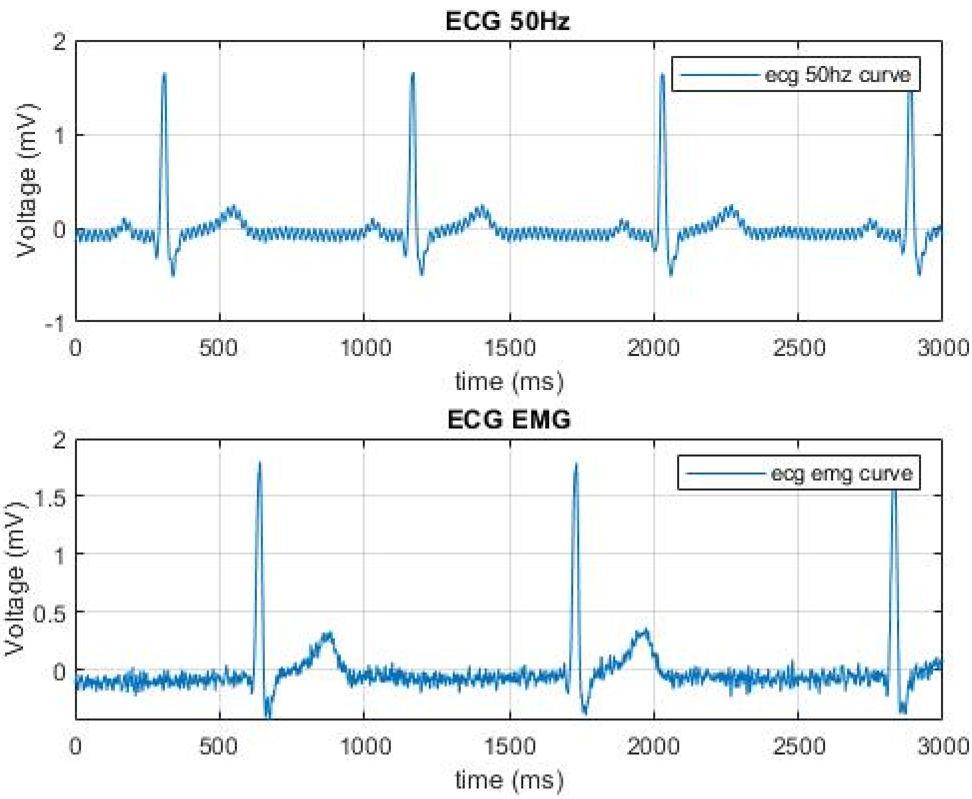
I:

a)

```
t=1:10000;
figure
subplot(2,1,1), plot(t(1:3000),ecg50hz(1:3000));
xlabel('time (ms)')
ylabel('Voltage (mV)')
grid
title('ECG 50Hz')
legend('ecg 50hz curve')
hold on
subplot(2,1,2), plot(t(1:3000),ecg_emg(1:3000));
grid
xlabel('time (ms)')
ylabel('Voltage (mV)')
title('ECG EMG')
legend('ecg emg curve')
```



II:

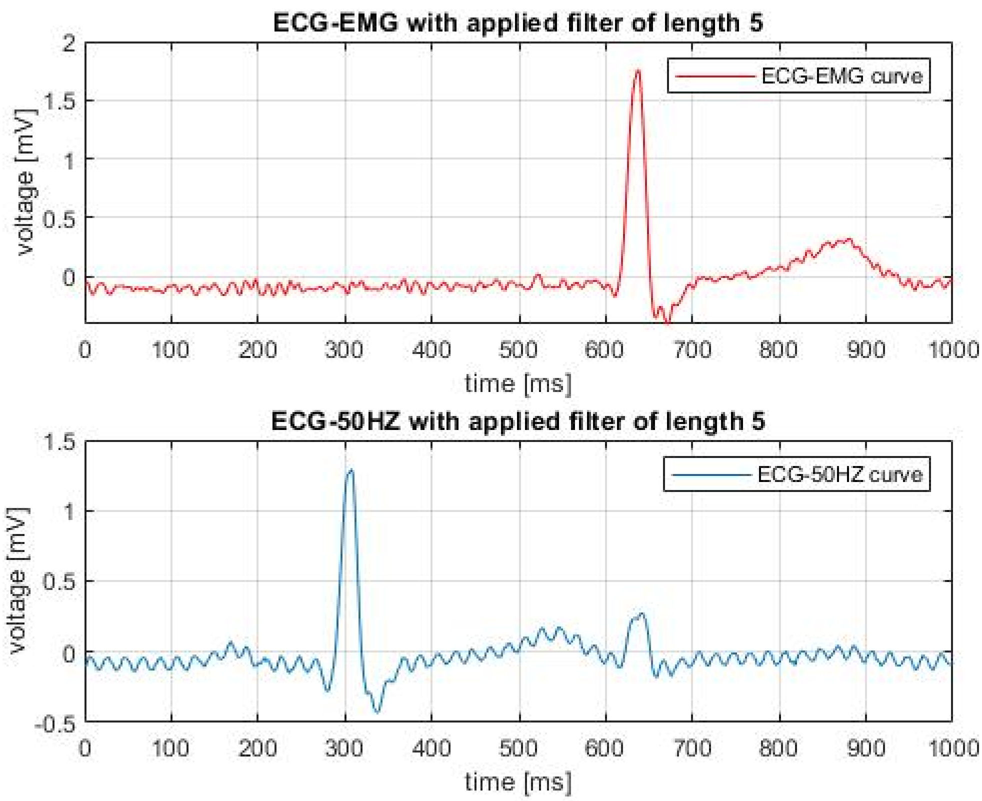
b)

```
n = 1:10000;

for i = 1:(length(n) - 4)
```

```
    v(i) = (ecg_emg(i) + ecg_emg(i+1) + ecg_emg(i+2) + ecg_emg(i+3) +
ecg_emg(i+4))/5;

end
```

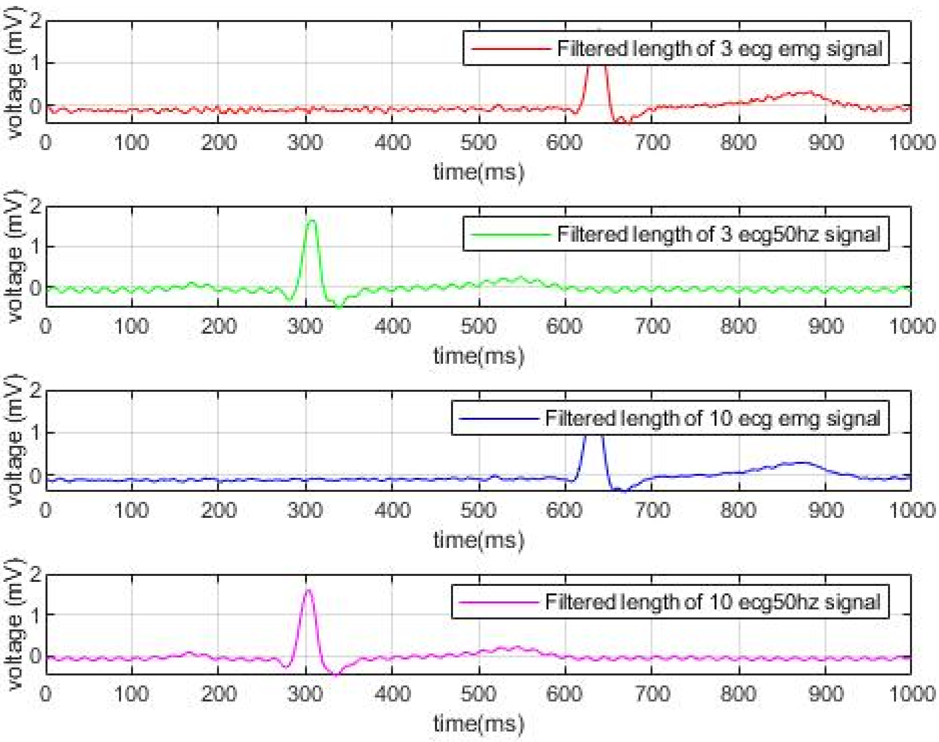
c)

```
n = 1:10000;

for l = 1:(length(n) - 4)

    e(l) = (ecg_emg(l) + ecg50hz(l+1) + ecg50hz(l+2) + ecg50hz(l+3) +
ecg50hz(l+4))/5;

end

figure
subplot(2,1,1), plot(t(1:1000),v(1:1000),'r-')
grid
xlabel('time [ms]')
ylabel('voltage [mV]')
title('ECG-EMG with applied filter of length 5')
legend('ECG-EMG curve')
hold on
subplot(2,1,2), plot(t(1:1000),e(1:1000))
grid
xlabel('time [ms]')
ylabel('voltage [mV]')
title('ECG-50HZ with applied filter of length 5')
legend('ECG-50HZ curve')
```



d)

```
n = 1:10000;

for d = 1:(length(n) - 3)

    x(d) = (ecg_emg(d) + ecg_emg(d+1) + ecg_emg(d+2))/3;

end

for g = 1:(length(n) - 3)

    z(g) = (ecg50hz(g) + ecg50hz(g+1) + ecg50hz(g+2))/3;

end

for j = 1:(length(n) - 10)

    h(j) = (ecg_emg(j) + ecg_emg(j+1) + ecg_emg(j+2)+ ecg_emg(j+3)+
ecg_emg(j+4)+ ecg_emg(j+5)+ ecg_emg(j+6)+ ecg_emg(j+7)+ ecg_emg(j+8)+
ecg_emg(j+9))/10;

end

for u = 1:(length(n) - 10)

    f(u) = (ecg50hz(u) + ecg50hz(u+1) + ecg50hz(u+2) + ecg50hz(u+3) +
ecg50hz(u+4) + ecg50hz(u+5) + ecg50hz(u+6) + ecg50hz(u+7) + ecg50hz(u+8) +
ecg50hz(u+9))/10;

end

figure
subplot(4,1,1), plot(t(1:1000),x(1:1000),'r-')
grid
xlabel('time(ms)')
ylabel('voltage (mV)')
legend('Filtered length of 3 ecg emg signal')
hold on
subplot(4,1,2), plot(t(1:1000),z(1:1000),'g-')
grid
xlabel('time(ms)')
ylabel('voltage (mV)')
legend('Filtered length of 3 ecg50hz signal')
hold on
subplot(4,1,3), plot(t(1:1000),h(1:1000),'b-')
grid
xlabel('time(ms)')
ylabel('voltage (mV)')
legend('Filtered length of 10 ecg emg signal')
hold on
subplot(4,1,4), plot(t(1:1000),f(1:1000),'m-')
grid
xlabel('time(ms)')
ylabel('voltage (mV)')
legend('Filtered length of 10 ecg50hz signal')
```



The effect of changing the length of filter is increase of values, due to the amount of values summed for one average operation. Also, the greater is the filtered length the smaller is the length of the averaged signal.

e) – Function

```
function[smooth_signal] = Mateusz_Grzybowski_function(a,b) % a - this
variable represents any arbitrary signal
                                                             % b - this
variable stands for the filter length
if b <= 1
    disp('The filter length you typed is wrong. Try again.') % This will
inform the user if he entered the filter length properly.
                                                              % Thus, it
solves an error which might occur.
    elseif length(a) <= 1
        disp('You entered a wrong length of the signal. Please enter some
other length of the signal') % The user is supposed to type in some domain
of the signal, not a single point.

% Hence, it informs user about a problem with his input.
    else
        for c = 1:(length(a) - (b-1)) %This is the for loop which computes
every signal with a chosen filter.
            A = [a(c);a(c+b-1)]; %This is to make the whole function more
clear.
            smooth_signal(c) = mean(A); %this operation evaluates average
for every next b numbers.
                %The output of the function is a vector of values which
                %is actually the a signal filtered with a length b
```

```
        end
    end

end
```

III:

f)

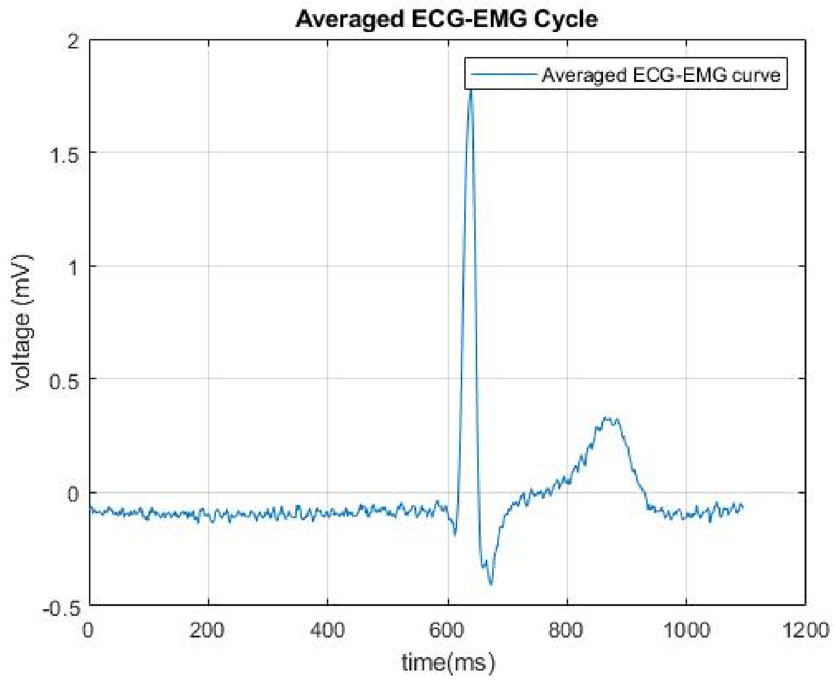
```
for s = 1:10000
    if ecg_emg(s) >= 1.7529
       disp(s)
    end
end
```

Results(all threshold points):

639	640	1731	1732	1733	2831	2832	2833	2834	3938	3939	5027	5028	5029	6112	6113	7157	7158	7159	8196	8197	8198	8199	9173

g)

```
a = 639;
b = 454; % counted values for a length of the cycle which is a+b
X = [640, 1733, 2834, 3939, 5029, 6113, 7159, 8199, 9173]; %Those are
threshold values which I have chosen for that task (i.e they are the last
threshold values for each cycle)

for r = 1:9
    sign_for(r,:) = ecg_emg(X(r) - a : X(r) + b);
end

k = mean(sign_for);

figure
plot(t(1:1094),k)
grid
xlabel('time(ms)')
ylabel('voltage (mV)')
legend('Averaged ECG-EMG curve')
title('Averaged ECG-EMG Cycle')
```



h)

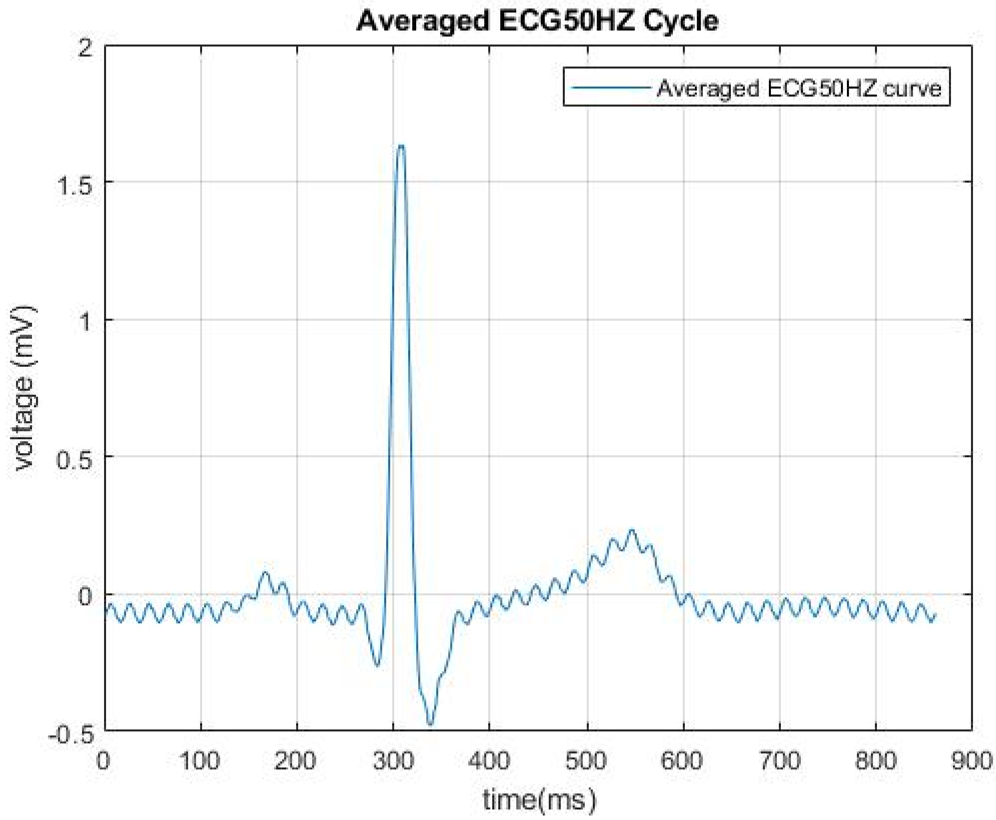
```
for w = 1:10000
    if ecg50hz(w) >= 1.1582
       disp(w)
    end
end
```

Results(all threshold points):

301		302		303		304		305		306		307		308		309		310		311		312		313		314		1161		1162	
1163		1164		1165		1166		1167		1168		1169		1170		1171		1172		1173		1174		2023		2024		2025		2026	
2027		2028		2029		2030		2031		2032		2033		2034		2035		2036		2037		2881		2882		2883		2884		2885	
2886		2887		2888		2889		2890		2891		2892		2893		2894		3802		3803		3804		3805		3806		3807		3808	
3809		3810		3811		3812		3813		3814		3815		3816		4737		4738		4739		4740		4741		4742		4743		4744	
4		4		4		4		4		4		4		5		5		5		5		5		5		5		5		5	

745		746		747		748		749		750		751		638		639		640		641		642		643		644		645		646	
5647		5648		5649		5650		5651		6565		6566		6567		6568		6569		6570		6571		6572		6573		6574		6575	
6576		6577		6578		6579		7501		7502		7503		7504		7505		7506		7507		7508		7509		7510		7511		7512	
7513		7514		8404		8405		8406		8407		8408		8409		8410		8411		8412		8413		8414		8415		8416		8417	
8418		9356		9357		9358		9359		9360		9361		9362		9363		9364		9365		9366		9367		9368		9369			

```
Q = [314, 1174, 2037, 2894, 3816, 4751, 5651, 6579, 7514, 8418,
9369]; %Those are threshold values which I have chosen for that task (i.e
they are the last threshold values for each cycle)
o = 313;
p = 547; % Counted values for a length of cycle which is o+p

for y = 1:11
    sign_forecg50hz(y,:) = ecg50hz(Q(y) - o : Q(y) + p);
end

m = mean(sign_forecg50hz);

figure
plot(t(1:861),m)
grid
xlabel('time(ms)')
ylabel('voltage (mV)')
legend('Averaged ECG50HZ curve')
title('Averaged ECG50HZ Cycle')
```



To sum up, comparing both methods of averaging cycles we are able to conclude some points:

1) First method is more accurate, because it sums every possible value to each-other and taking appropriate average of it whilst in the second method some of the points are avoided or calculated twice. This factor might have an impact on the final results of the graph. Fortunately, those differences are not that significant.

2) The first method is way faster to write and apply than the second one (in my opinion).

3) For ECG50HZ signal, using second method of averaging keeps values more similar to the original curve comparing to the first method. For sure, second method is better here.

4) However, for ECG_EMG signal both methods seem to present nearly the same curve which are similar to the original one. Thus, we cannot say whether 1st or 2nd are better here.

Conclusion: 1st method is significantly faster in writing and usage than 2nd method. On the other hand, 2nd method is personally speaking more accurate. If I had to choose which method is better, I think I would choose 2nd one owing to that accuracy is in most cases more important than saving some time on doing something less accurate.