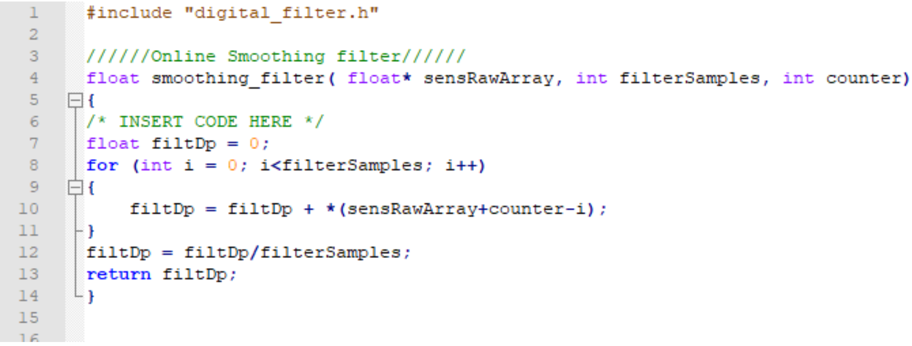
Introduction to Robotics and Mechatronics

GROUP 2.5

Benson Chalethu
Pietro Griffa
Ingvar Groza

PreLab 04

Q1



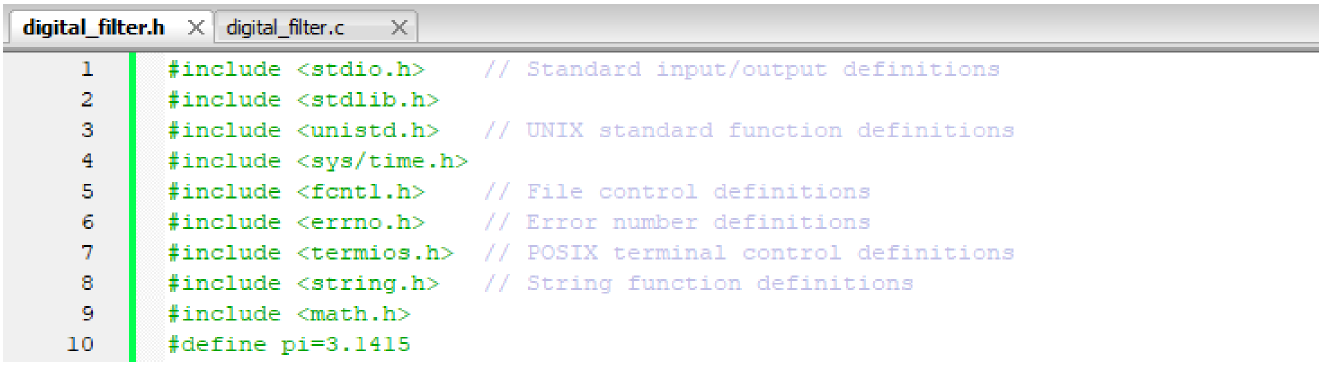
```c
#include "digital_filter.h"

//////Online Smoothing filter//////
float smoothing_filter( float* sensRawArray, int filterSamples, int counter)
{
/* INSERT CODE HERE */
float filtDp = 0;
for (int i = 0; i<filterSamples; i++)
{
    filtDp = filtDp + *(sensRawArray+counter-i);
}
filtDp = filtDp/filterSamples;
return filtDp;
}
```

Q2

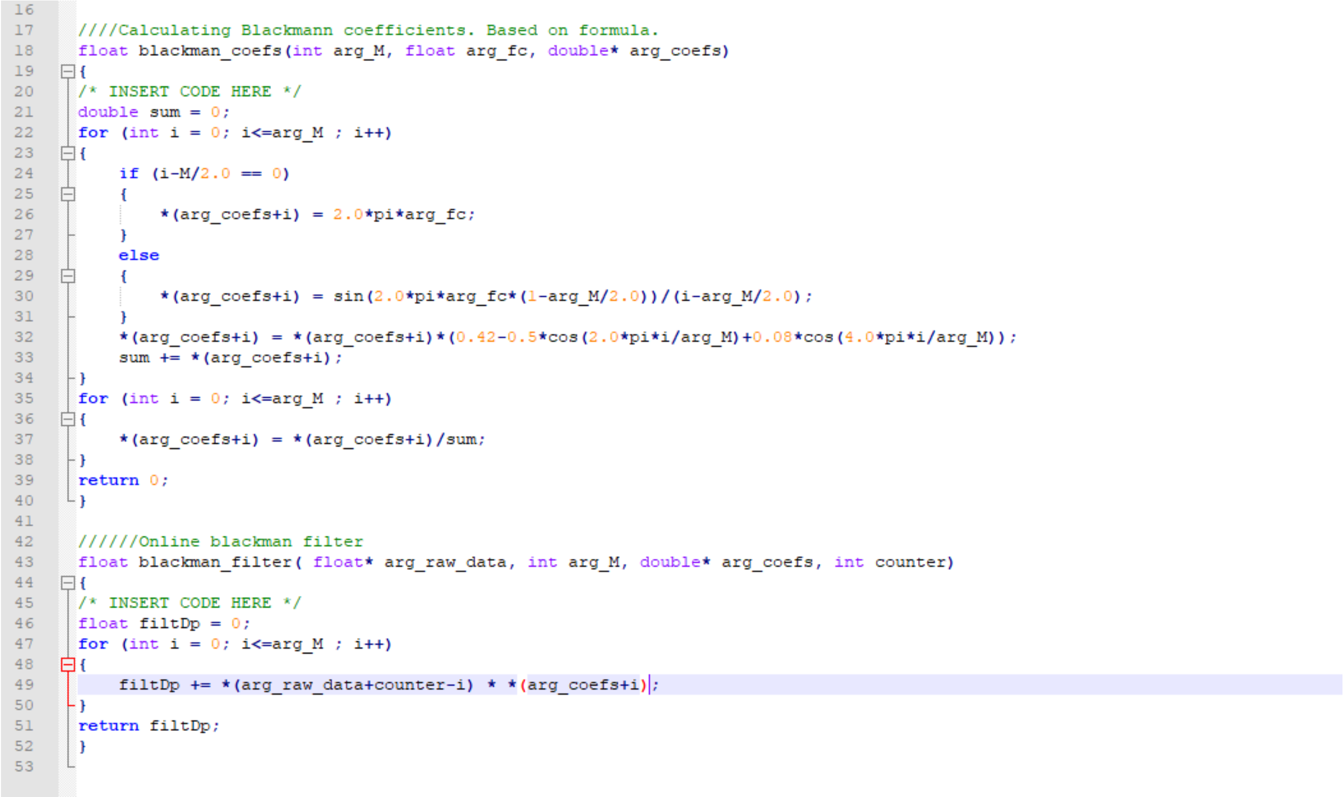
$$M = 200$$
$$f_c = 0{,}1$$

Q3



digital_filter.h   digital_filter.c

```c
#include <stdio.h>     // Standard input/output definitions
#include <stdlib.h>
#include <unistd.h>    // UNIX standard function definitions
#include <sys/time.h>
#include <fcntl.h>     // File control definitions
#include <errno.h>     // Error number definitions
#include <termios.h>   // POSIX terminal control definitions
#include <string.h>    // String function definitions
#include <math.h>
#define pi=3.1415
```



```c

////Calculating Blackmann coefficients. Based on formula.
float blackman_coefs(int arg_M, float arg_fc, double* arg_coefs)
{
/* INSERT CODE HERE */
double sum = 0;
for (int i = 0; i<=arg_M ; i++)
{
    if (i-M/2.0 == 0)
    {
        *(arg_coefs+i) = 2.0*pi*arg_fc;
    }
    else
    {
        *(arg_coefs+i) = sin(2.0*pi*arg_fc*(1-arg_M/2.0))/(i-arg_M/2.0);
    }
    *(arg_coefs+i) = *(arg_coefs+i)*(0.42-0.5*cos(2.0*pi*i/arg_M)+0.08*cos(4.0*pi*i/arg_M));
    sum += *(arg_coefs+i);
}
for (int i = 0; i<=arg_M ; i++)
{
    *(arg_coefs+i) = *(arg_coefs+i)/sum;
}
return 0;
}

//////Online blackman filter
float blackman_filter( float* arg_raw_data, int arg_M, double* arg_coefs, int counter)
{
/* INSERT CODE HERE */
float filtDp = 0;
for (int i = 0; i<=arg_M ; i++)
{
    filtDp += *(arg_raw_data+counter-i) * *(arg_coefs+i);
}
return filtDp;
}

```