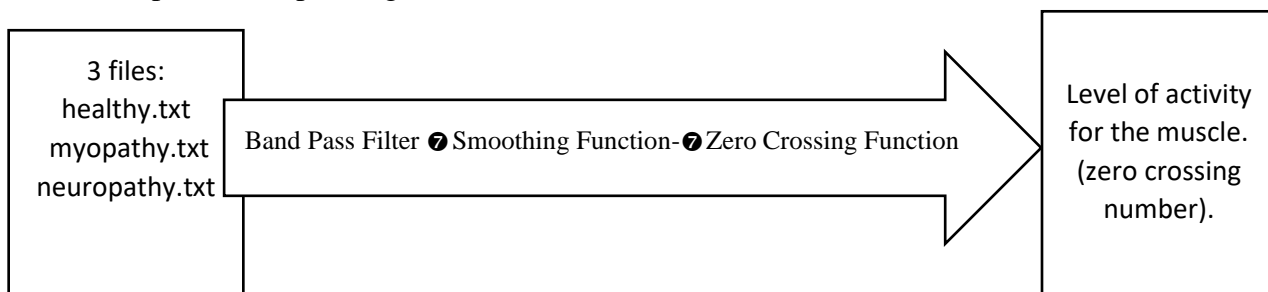


Problem Identification and Statement

Develop a program that will process EMG data before it can be used for analysis. EMG data includes noises that can be removed. The data should be noise filtered, rectified, smoothed, and normalized. In this project we are developing a program that reads the data from a file, and then perform filtration, rectification, smoothing, and normalization. At the end the software will display the level of activity of the muscles on the outer screen.

Gathering Information and Input/output diagram.

- **Band pass filter:** It is a filter that is used to remove low and high frequencies from a signal or from the data and accepts frequency signal around 60 Hz. This function will receive EMG data, low frequency, high frequency, sampling rate, and the size of the EMG data that we have and in return it will give us EMG array with filtered data.
The equation that this function uses are
1. $y_h[i] = \alpha * (y_h[i-1] + x[i] - x[i-1])$
where $\alpha = RC / (RC + dt)$;
 $RC = 1 / (2 * \pi * f_l)$ $y_h[0] = x[0]$ $f_l = 55\text{Hz}$
 $dt = 1 / \text{samplingrate}(4000)$
2. $y_l[i] := y_l[i-1] + \alpha * (x[i] - y_l[i-1])$
where $\alpha = dt / (RC + dt)$;
 $RC = 1 / (2 * \pi * f_h)$ $y_l[0] = x[0]$ $f_h = 65\text{Hz}$
 $dt = 1 / \text{samplingrate}(4000)$
- **Smoothing Function:** It is a function that gives us the average of a number and the three numbers that are following it. This function can be designed in different ways, one way is to make it read the data from the end then we use the following equation.
 $EMGdata[i] = (EMGdata[i] + EMGdata[i - 1] + EMGdata[i - 2] + EMGdata[i - 3]) / 4$;
- **Zero crossing:** When we get the data from smoothing, the zero crossing function counts how many times the x line is being crossed, it checks the sign of neighboring data points and counts the cases where the signs are different from each other. The equation used is $EMGdata[i] * EMGdata[i - 1] < 0$ where if two sets of arrays next to each other were multiplied and the answer is negative then the zero line was crossed because we had a positive number and a negative number.
- **Input and output diagram:**



Healthy.txt stores EMGdata for a 44 year old man without history of neuromuscular disease.

Neuropathy.txt stores EMG data for a 62 year old man with chronic low back pain and neuropathy due to a right L5 radiculopathy.

Myopathy.txt stores EMG data for a 57 year old man with myopathy due to longstanding history of polymyositis, treated effectively with steroids and low-dose methotrexate.

Test Cases and algorithm

1. Test Case 1

If the user input file name healthy.txt, he/she will have zero crossing of 193.

EMG Signal Processing

Please enter the file name where the EMG data is stored healthy.txt

The number of data points is:4000

The number of zero crossing is equal to: 193 Press

any key to continue . . .

2. Test Case 2

If the use input filename myopathy.txt he/she will have zero crossing of 348.

EMG Signal Processing

Please enter the file name where the EMG data is stored myopathy.txt

The number of data points is:4000

The number of zero crossing is equal to: 348 Press

any key to continue . . .

3. Test Case 3

If the use input filename neuropathy.txt he/she will have zero crossing of 244.

EMG Signal Processing

Please enter the file name where the EMG data is stored neuropathy.txt

The number of data points is:4000

The number of zero crossing is equal to: 244 Press

any key to continue . . .

4. Test Case 4

If the use input a file other than the three above, she/he will get the following message “Error opening the filePress any key to continue . . .” EMG

Signal Processing

Please enter the file name where the EMG data is stored emgdata.txt

Error opening the filePress any key to continue . . .

Algorithm

Set pi to 3.14 Set

SIZE to 4000

main program

Declare fl as an integer and assign 55 to it, declare fh as an integer and assign 65 to it

Declare dt as a double and assign 1.0/4000.0 to it

Print "EMG Signal is Processing"

Print "Please enter the file name where EMG data is stored"

Read in the filename

Declare size as an integer and assign LoadData (EMGdata, filename) to it

Call function: BandPassFilter(EMGdata, fl, fh, dt, size)

Call function: ApplySmoothing(EMGdata, size)

Declare num as an integer and assign ZeroCrossing(EMGdata, size) to it.

Print "The number of data points is" then the variable size and new line

Print "The number of zero crossing is equal to" then the variable num and new line

Return 0

LoadData(EMGdata array, filename): int

Declare time as double

Declare size as an integer and assign 0 to it

Create input file stream inFile

Open file "filename" into stream inFile for reading

If opening the file fails

PrintError "Error opening the file", Newline

Exit program

Repeat while inFile is not at the END OF FILE

Read data from file into time and then into EMGdata[size]

Increment size

Close file

Return size

BandPassFilter(EMGdata array, fl, fh, dt, size): void

Declare yh[SIZE], yl[SIZE], output[SIZE] as doubles

*Declare R1 as double and assign $1 / (2 * \pi * fl)$ to it, declare a1 as double and assign $R1 / (R1 + dt)$,*

*Declare R2 as double and assign $1 / (2 * \pi * fh)$ to it, declare a2 as a double and assign $dt / (R2 + dt)$ to it*

Assign EMGdata[0] to yh[0]

Assign EMGdata[0] to yl[0]

Assign 1 to i

Repeat while i is less than size

*Assign $a1 * (yh[i - 1] + EMGdata[i] - EMGdata[i - 1])$ to yh[i]*

Increment i

Assign 1 to i

Repeat while i is less than size

*Assign $yl[i - 1] + a2 * (EMGdata[i] - yl[i - 1])$ to yl[i]*

Increment i

Assign 0 to x

Repeat while x is less than size

Assign $yh[x] - yl[x]$ to output[x]

Assign output[x] to EMGdata[x]

ApplySmoothing(EMGdata array, size):void

Assign size-1 to i repeat while

i is bigger than 3

$EMGdata[i] = (EMGdata[i] + EMGdata[i - 1] + EMGdata[i - 2] + EMGdata[i - 3]) / 4;$

decrement i

ZeroCrossing(EMGdata array, size):int

Declare num as an integer and assign 0 to it.

Assign 1 to i

repeat while i is less than size

*If $EMGdata[i] * EMGdata[i - 1]$ is less than 0 then*

increment num

Increment i

Code

```

#include <iostream>
#include <fstream>
#include <string>
#define pi 3.14 #define
SIZE 4000 using
namespace std;
int LoadData(double EMGdata[], string filename);
void BandPassFilter(double EMGdata[], double fl, double fh, double dt, int size);
void ApplySmoothing(double EMGdata[], int size); int
ZeroCrossing(double EMGdata[], int size);
    int
main()
{
    //declaring and initilizing variables
    int fl = 55, fh = 65;
double dt = 1.0 / 4000.0;    string
filename;
    //printing the following sentences
cout << "EMG Signal Processing" << endl;
    cout << "Please enter the file name where the EMG data is stored" << endl;
    //the name of file stored in the filename
    cin >> filename;
//declaring array    double
EMGdata[SIZE];        //calling
functions
    int size = LoadData(EMGdata, filename);
BandPassFilter(EMGdata, fl, fh, dt, size);
ApplySmoothing(EMGdata, size);    int num =
ZeroCrossing(EMGdata, size);
    //printing size
    cout << "The number of data points is:" << size << endl;
    //printing output
    cout << "The number of zero crossing is equal to: " << num << endl;
system("pause");
    return(0);
}
// function that load the data into an array for further processing
int LoadData(double EMGdata[], string filename) {
    //declare and initialization
    double time;
int size = 0;
//opening file
ifstream infile;
    //checking if the files open or
not    infile.open(filename, ios::in);
if (infile.fail())
{
    //print message if the file is not opening
    cerr << "Error opening the
file";
    system("pause");
exit(1);
}
    //do while loop until the END OF FILE
while (!infile.eof())
{
    //    Reading data from file into time and then into EMGdata[size]
    infile >> time >> EMGdata[size];

```

```

        size++;
    }
    infile.close();
    //returning size to the data
    return size;
}
//Function that filters data and remove low and high frequencies from a signal
void BandPassFilter(double EMGdata[], double fl, double fh, double dt, int size) {
    //declaring arrays
    double yh[SIZE], yl[SIZE], output[SIZE];
    //declaring variables and assigning equations for them
    double R1 = 1 / (2 * pi * fl), a1 = R1 / (R1 + dt), R2 = 1 / (2 * pi * fh), a2 = dt / (R2
+ dt);
    //Assigning EMGdata
    yh[0] = EMGdata[0];
    yl[0] = EMGdata[0];
    //loop for high pass filter equation
    for (int i = 1; i < size; i++)
    {
        yh[i] = a1 * (yh[i - 1] + EMGdata[i] - EMGdata[i - 1]);
    }
    //loop for low pass filter equation
    for (int i = 1; i < size; i++)
    {
        yl[i] = yl[i - 1] + a2 * (EMGdata[i] - yl[i - 1]);
    }
    //loop for band pass filter equation
    for (int x = 0; x < size; x++)
    {
        output[x] = yh[x] - yl[x];
        EMGdata[x] = output[x];
    }
}

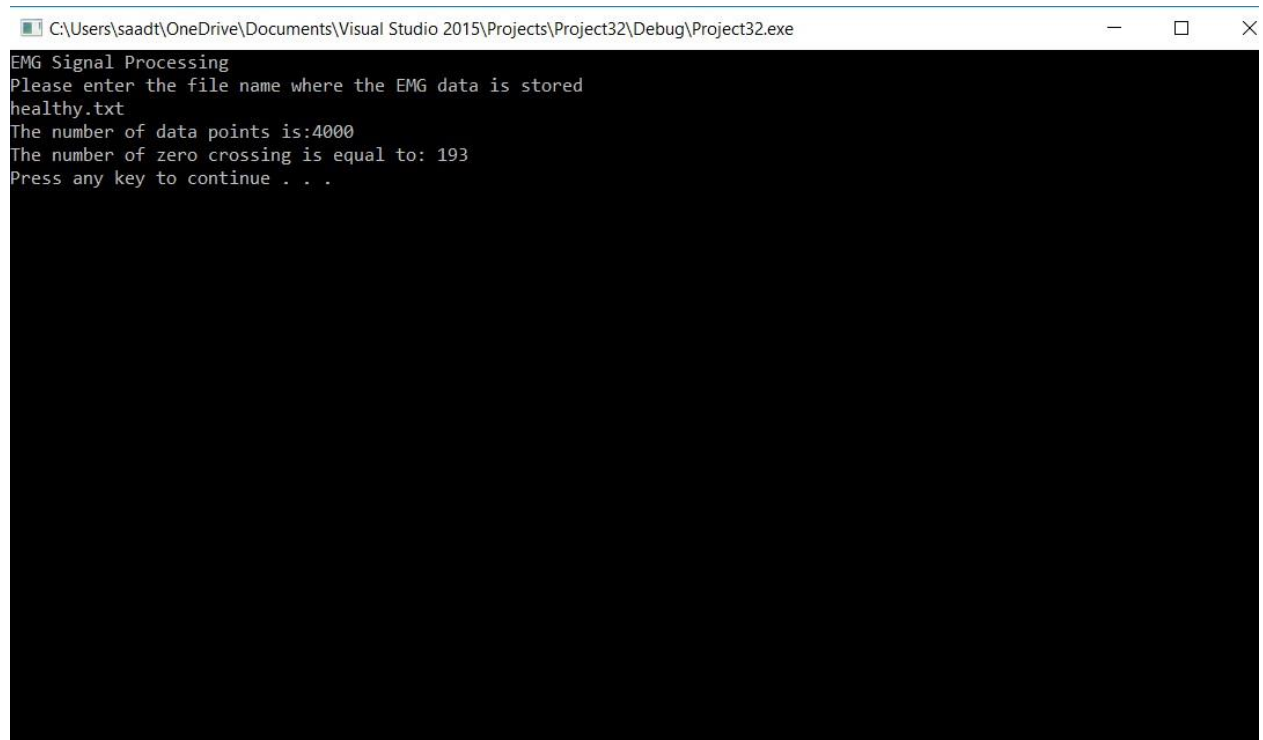
// function that gives us the average of a number and the three numbers that are
following it
void ApplySmoothing(double EMGdata[], int size)
{
    //loop for Apply Smoothing equation
    for (int i = size - 1; i > 3; i--)
    {
        EMGdata[i] = (EMGdata[i] + EMGdata[i - 1] + EMGdata[i - 2] + EMGdata[i -
3]) / 4;
    }
}
// function that shows us how many time the line was crossed int
ZeroCrossing(double EMGdata[], int size)
{
    int num = 0;
    {
        for (int i = 1; i < size; i++)
        // if two sets of arrays next to each other were multiplied and the answer is negative
then the zero line was crossed because we had a positive number and a negative number.
        if (EMGdata[i] * EMGdata[i - 1] < 0)
            num++;
    }
    return num;
}

```

}

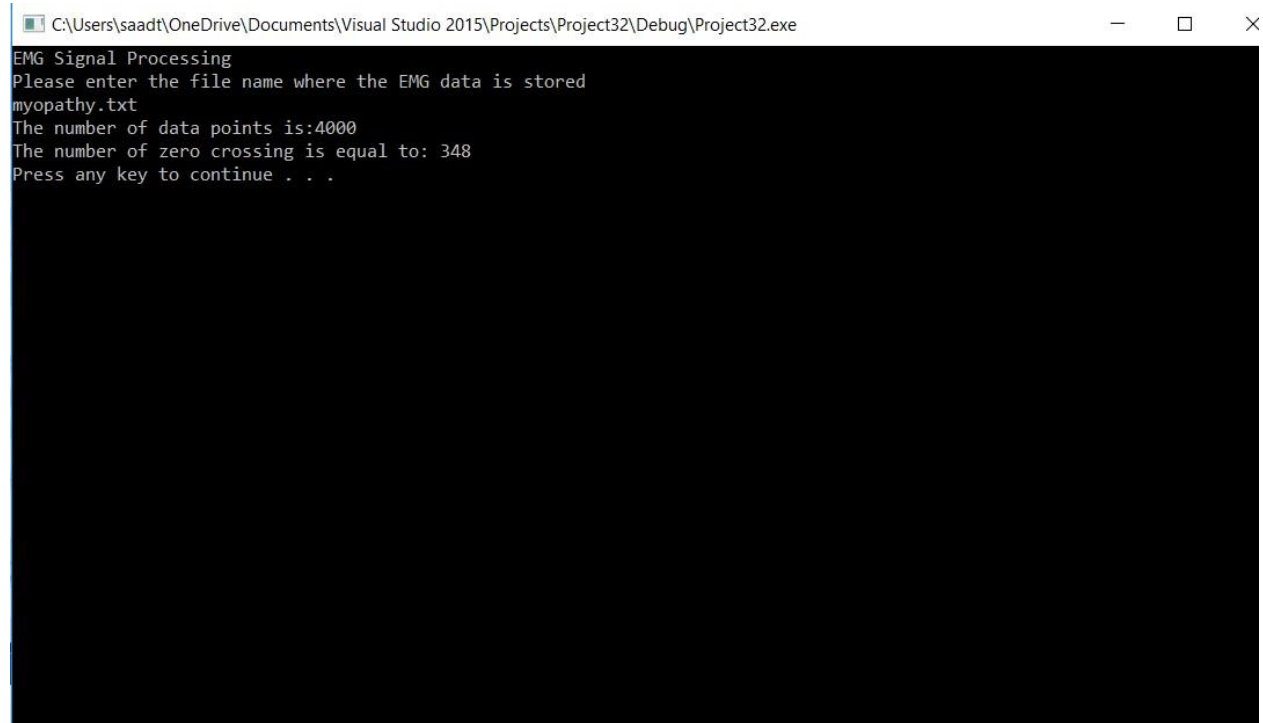
Test Cases:

Test Case 1



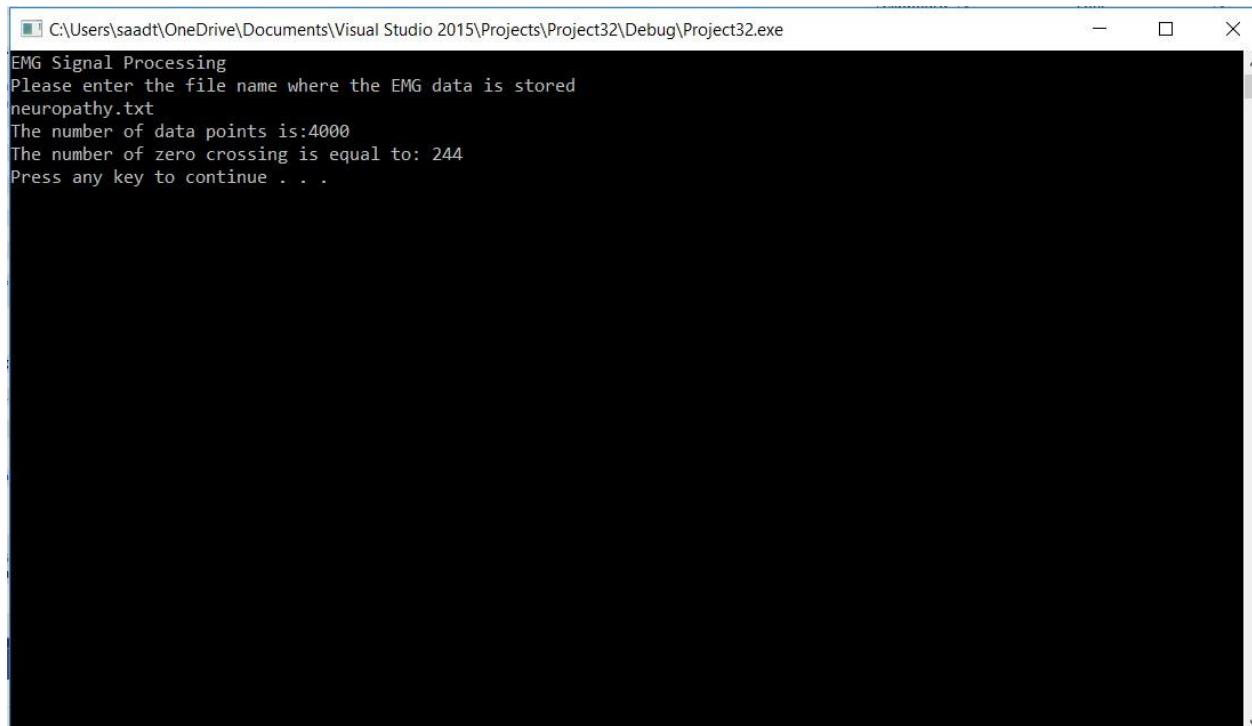
```
C:\Users\saadt\OneDrive\Documents\Visual Studio 2015\Projects\Project32\Debug\Project32.exe
EMG Signal Processing
Please enter the file name where the EMG data is stored
healthy.txt
The number of data points is:4000
The number of zero crossing is equal to: 193
Press any key to continue . . .
```

Test Case 2



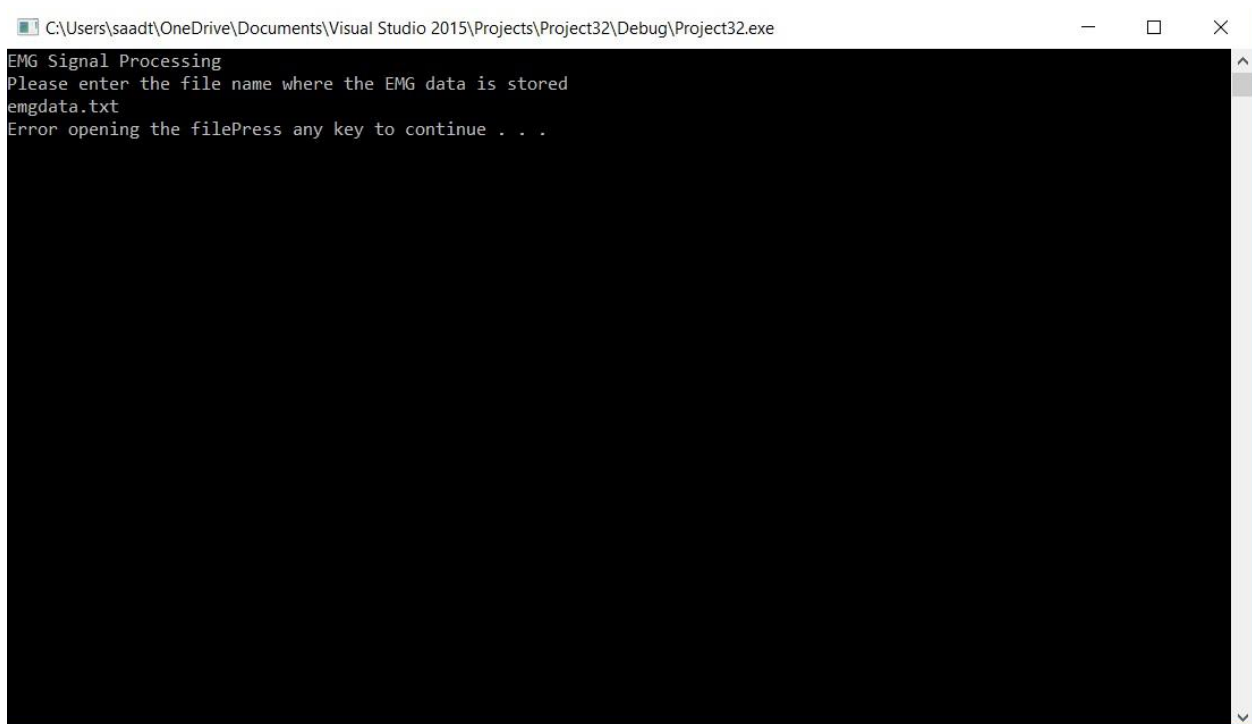
```
C:\Users\saadt\OneDrive\Documents\Visual Studio 2015\Projects\Project32\Debug\Project32.exe
EMG Signal Processing
Please enter the file name where the EMG data is stored
myopathy.txt
The number of data points is:4000
The number of zero crossing is equal to: 348
Press any key to continue . . .
```


Test Case 3



```
C:\Users\saadt\OneDrive\Documents\Visual Studio 2015\Projects\Project32\Debug\Project32.exe
EMG Signal Processing
Please enter the file name where the EMG data is stored
neuropathy.txt
The number of data points is:4000
The number of zero crossing is equal to: 244
Press any key to continue . . .
```

Test Case 4



```
C:\Users\saadt\OneDrive\Documents\Visual Studio 2015\Projects\Project32\Debug\Project32.exe
EMG Signal Processing
Please enter the file name where the EMG data is stored
emgdata.txt
Error opening the filePress any key to continue . . .
```