MatLab Project

An interface for numbers for performing the following functions in MATLAB

1. To check whether the given number is prime or not

2. TO check the equivalent resistance of 2 parallel combination of Resistors

3. To plot a frequency curve of choice made

4. To find and display the sum of 10 terms of a Geometric Progression [GP]

Group Members

1. Abhijith

2.Abhinav

3.Aamir

4.Abhishek

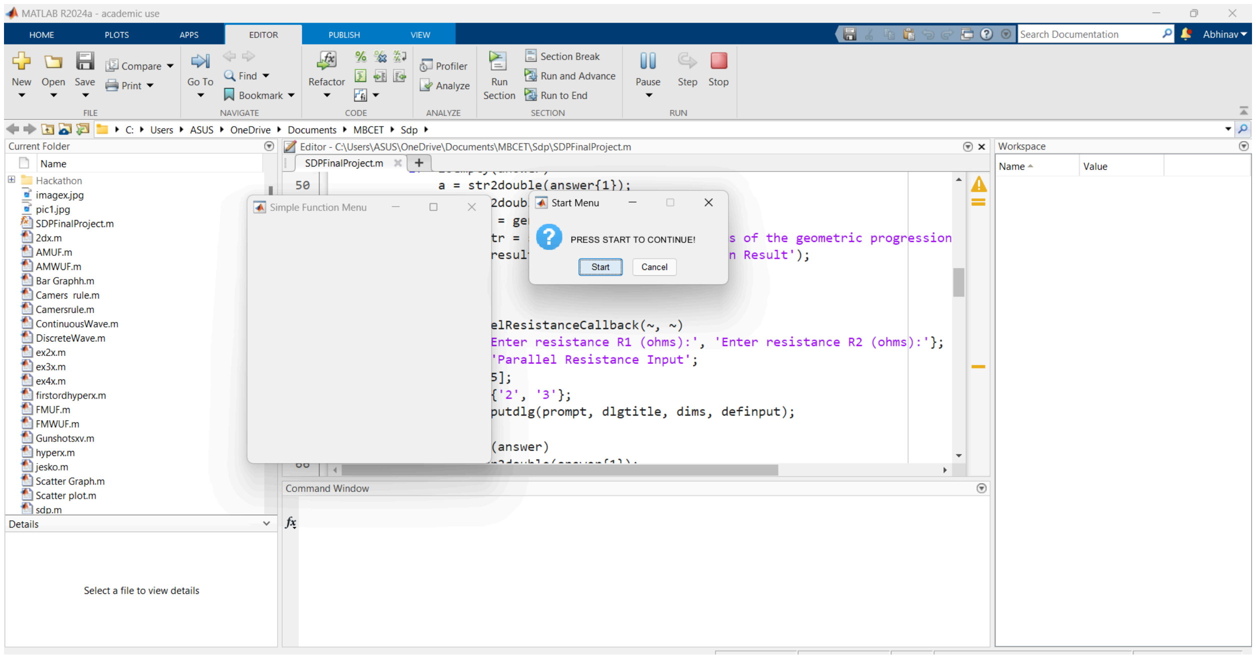
Code :

```
function MainMenu()
clc;
clear all;
close all;
   fig = figure('Position', [300, 300, 300, 300], 'MenuBar', 'none', ...
                'Name', 'Simple Function Menu', 'NumberTitle', 'off', ...
                'CloseRequestFcn', @closeGui);
   startMenu();
   function startMenu()
       selection = questdlg('PRESS START TO CONTINUE!', ...
           'Start Menu', ...
           'Start', 'Cancel', 'Start');
       switch selection
           case 'Start'
               createMainMenu();
           case 'Cancel'
               closeGui();
       end
   end
   function createMainMenu()
       uicontrol('Style', 'pushbutton', 'String', 'Geometric Progression', ...
                 'Position', [50, 220, 200, 30], 'Callback', @gpCallback);
       uicontrol('Style', 'pushbutton', 'String', 'Parallel Resistance', ...
                 'Position', [50, 180, 200, 30], 'Callback',
@parallelResistanceCallback);
       uicontrol('Style', 'pushbutton', 'String', 'Check Prime', ...
                 'Position', [50, 140, 200, 30], 'Callback',
@primeCheckerCallback);
       uicontrol('Style', 'pushbutton', 'String', 'Plot Frequency Curve', ...
                 'Position', [50, 100, 200, 30], 'Callback',
@plotFrequencyCallback);
   end
   function closeGui(~, ~)
       delete(fig);
   end
   function gpCallback(~, ~)
       prompt = {'Enter the first term (a):', 'Enter the common ratio (r):'};
       dlgtitle = 'Geometric Progression Input';
       dims = [1 35];
       definput = {'3', '2'};
       answer = inputdlg(prompt, dlgtitle, dims, definput);

       if ~isempty(answer)
           a = str2double(answer{1});
           r = str2double(answer{2});
           gpTerms = generateGP(a, r);
```

```
            resultStr = sprintf('The first ten terms of the geometric
progression are:\n%s', num2str(gpTerms));
            msgbox(resultStr, 'Geometric Progression Result');
        end
    end
    function parallelResistanceCallback(~, ~)
        prompt = {'Enter resistance R1 (ohms):', 'Enter resistance R2 (ohms):'};
        dlgtitle = 'Parallel Resistance Input';
        dims = [1 35];
        definput = {'2', '3'};
        answer = inputdlg(prompt, dlgtitle, dims, definput);

        if ~isempty(answer)
            R1 = str2double(answer{1});
            R2 = str2double(answer{2});
            result = parallelResistanceCalc(R1, R2);
            msgbox(['Equivalent resistance: ', num2str(result), ' ohms'],
'Parallel Resistance Result');
        end
    end
    function primeCheckerCallback(~, ~)
        prompt = {'Enter a number to check if it is prime:'};
        dlgtitle = 'Prime Number Check';
        dims = [1 35];
        definput = {'2'};
        answer = inputdlg(prompt, dlgtitle, dims, definput);

        if ~isempty(answer)
            numToCheck = str2double(answer{1});
            result = primechecker(numToCheck);
            if result
                msgbox([num2str(numToCheck), ' is a prime number.'], 'Prime
Number Result');
            else
                msgbox([num2str(numToCheck), ' is not a prime number.'], 'Prime
Number Result');
            end
        end
    end
    function plotFrequencyCallback(~, ~)
        prompt = {'Enter your frequency data in kHz (use square brackets for
array format):'};
        dlgtitle = 'Frequency Curve Input';
        dims = [1 35];
        definput = {'[1, 2, 1.5, 3, 2.5, 3.5, 2, 1.8, 1.2, 3]'};
        answer = inputdlg(prompt, dlgtitle, dims, definput);

        if ~isempty(answer)
            data_khz = str2num(answer{1});
            plotFrequencyCurve(data_khz);
            msgbox('Frequency curve plotted successfully!', 'Plot Result');
        end
    end
end
function gpTerms = generateGP(a, r)
    gpTerms = a * (r.^(0:9));
end
function eqresistance = parallelResistanceCalc(R1, R2)
    eqresistance = (R1 * R2) / (R1 + R2);
end
function isPrime = primechecker(num)
    if num <= 1
        isPrime = false;
        return;
    end
    for i = 2:sqrt(num)
        if mod(num, i) == 0
            isPrime = false;
            return;
        end
    end
    isPrime = true;
end
function plotFrequencyCurve(data_khz)
    [freq, edges] = histcounts(data_khz, 'Normalization', 'probability');
    binCenters = edges(1:end-1) + diff(edges)/2;
    figure;
    plot(binCenters, freq, '-o', 'LineWidth', 2);
    title('Frequency Curve of Input Data in kHz');
    xlabel('Frequency (kHz)');
    ylabel('Probability Density');
    grid on;
end
```

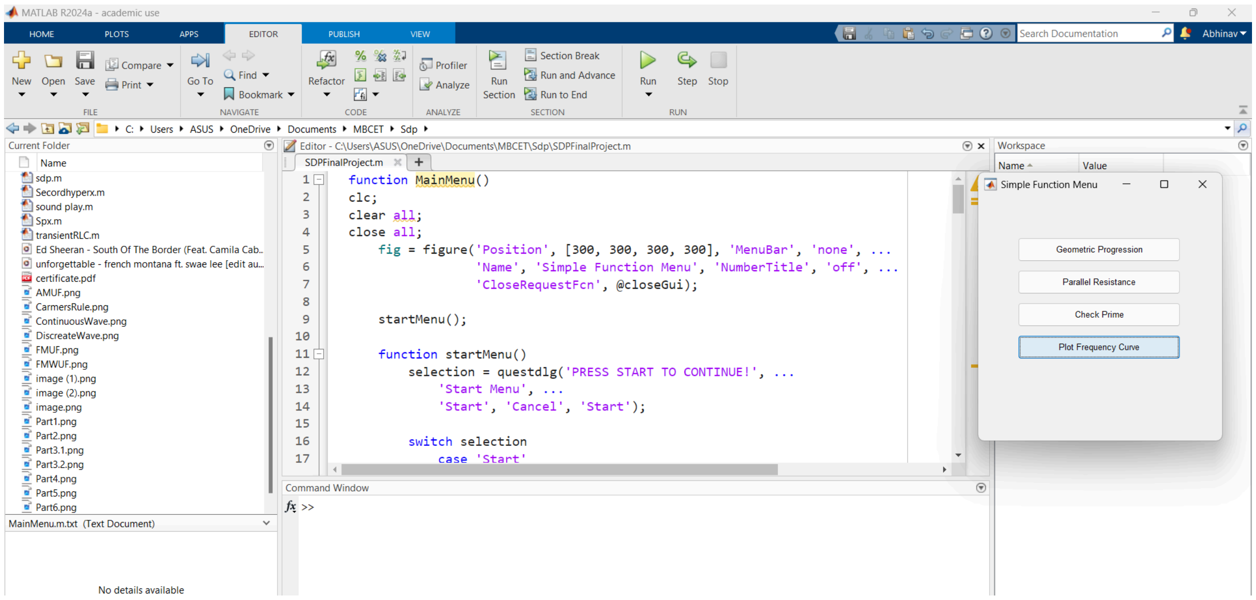
Working :

Phase 1:



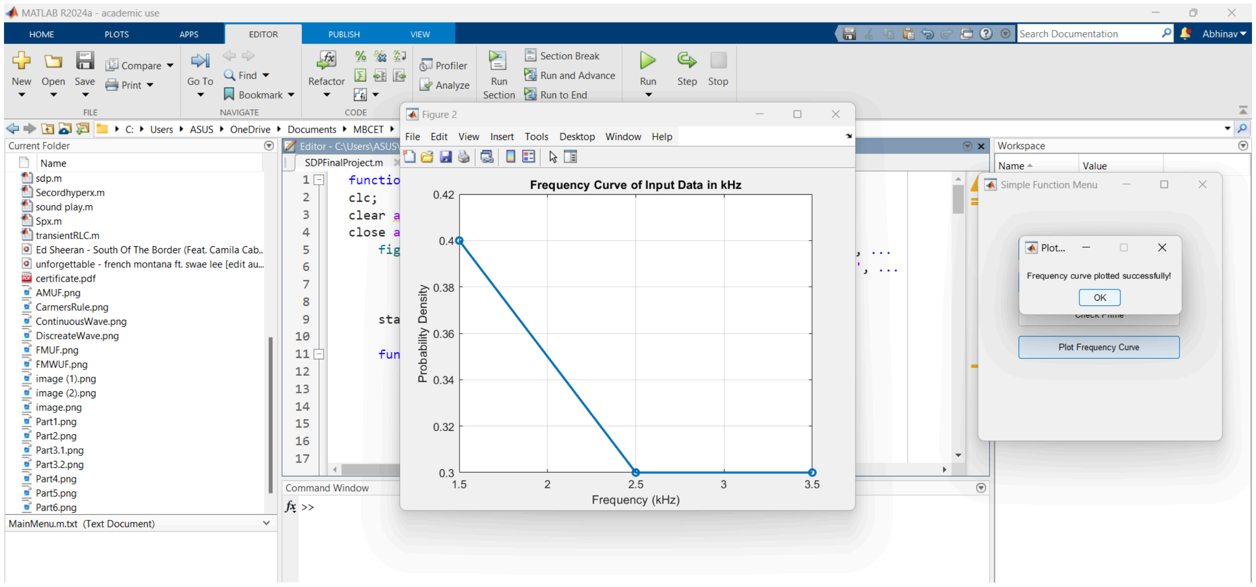
```
            a = str2double(answer{1});
        2doub
        = ge
        tr =                                   s of the geometric progression
        resul                                  n Result');



elResistanceCallback(~, ~)
Enter resistance R1 (ohms):', 'Enter resistance R2 (ohms):'};
'Parallel Resistance Input';
5];
{'2', '3'};
putdlg(prompt, dlgtitle, dims, definput);

(answer)
```

Phase 2:



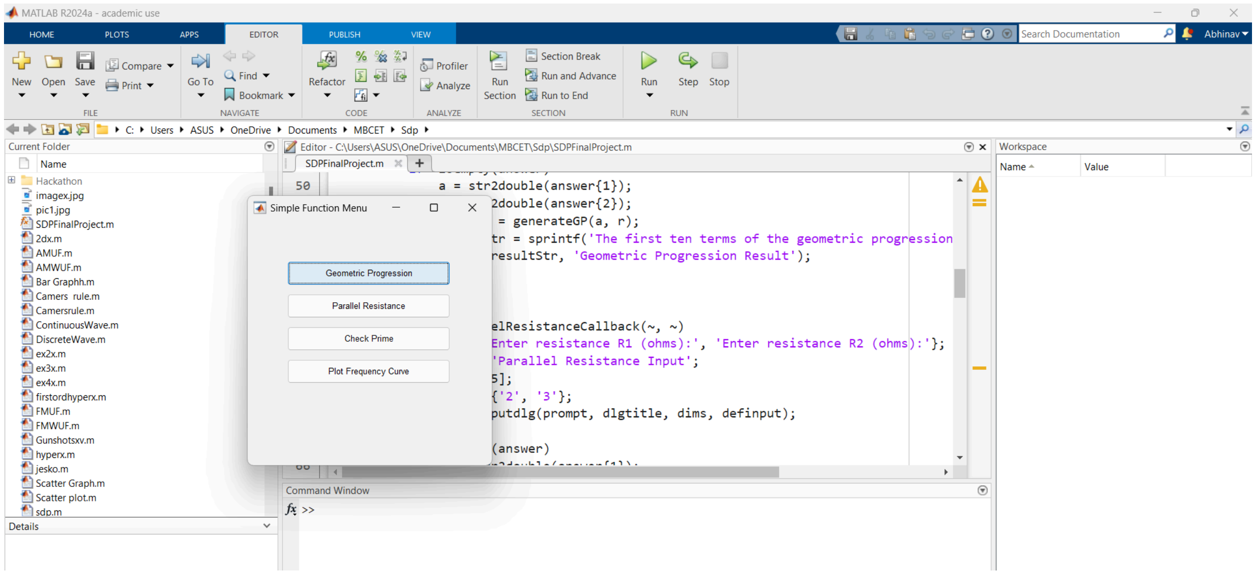
```matlab
function MainMenu()
clc;
clear all;
close all;
    fig = figure('Position', [300, 300, 300, 300], 'MenuBar', 'none', ...
                 'Name', 'Simple Function Menu', 'NumberTitle', 'off', ...
                 'CloseRequestFcn', @closeGui);

    startMenu();

    function startMenu()
        selection = questdlg('PRESS START TO CONTINUE!', ...
            'Start Menu', ...
            'Start', 'Cancel', 'Start');

        switch selection
            case 'Start'
```

Output:



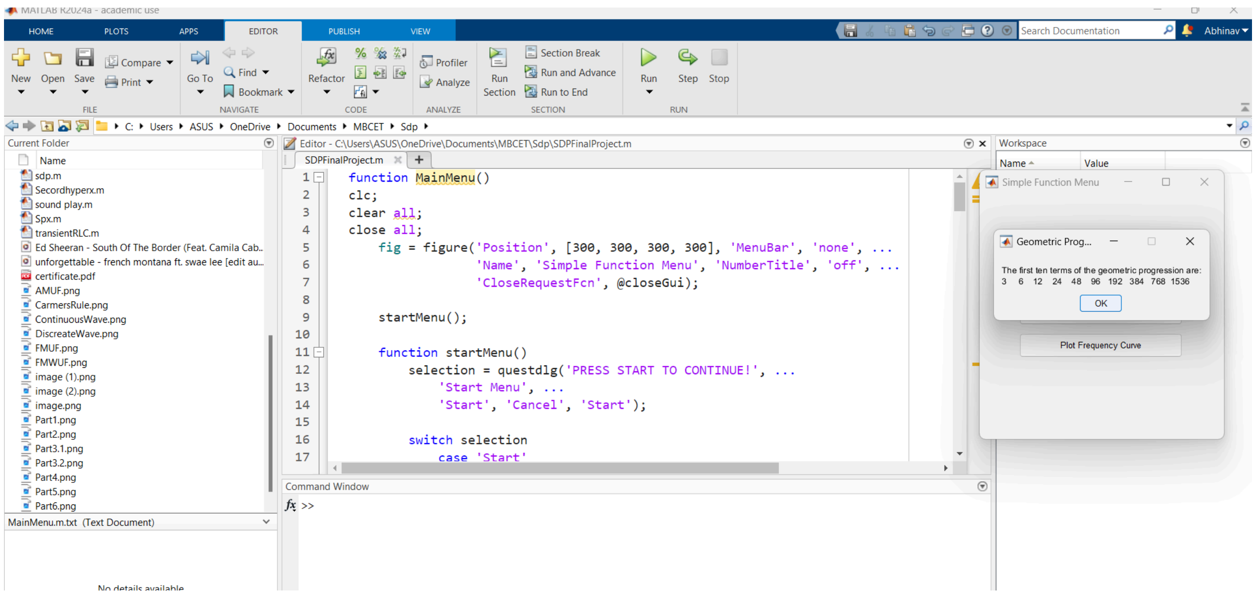
Frequency curve plotted successfully!

Phase 3:



```
            a = str2double(answer{1});
 2double(answer{2});
  = generateGP(a, r);
tr = sprintf('The first ten terms of the geometric progression
resultStr, 'Geometric Progression Result');



elResistanceCallback(~, ~)
Enter resistance R1 (ohms):', 'Enter resistance R2 (ohms):'};
'Parallel Resistance Input';
5];
{'2', '3'};
putdlg(prompt, dlgtitle, dims, definput);

(answer)
```

Output :

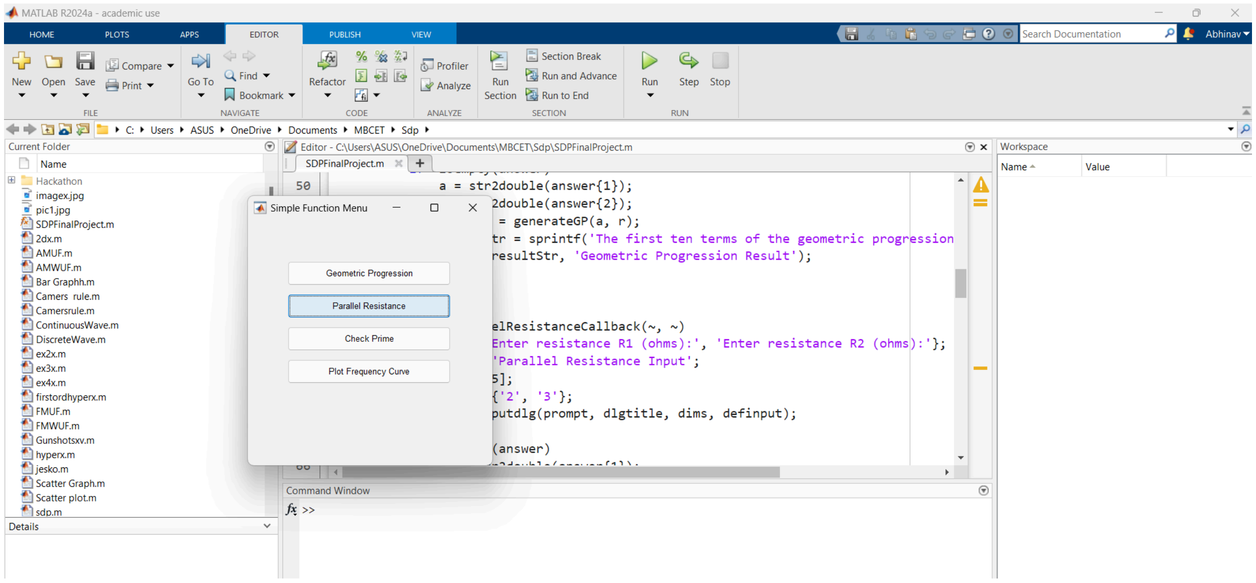


MATLAB R2024a - academic use
Editor - C:\Users\ASUS\OneDrive\Documents\MBCET\Sdp\SDPFinalProject.m
Simple Function Menu
Geometric Prog...
OK
Plot Frequency Curve
Command Window

```
function MainMenu()
clc;
clear all;
close all;
    fig = figure('Position', [300, 300, 300, 300], 'MenuBar', 'none', ...
                 'Name', 'Simple Function Menu', 'NumberTitle', 'off', ...
                 'CloseRequestFcn', @closeGui);

    startMenu();

    function startMenu()
        selection = questdlg('PRESS START TO CONTINUE!', ...
            'Start Menu', ...
            'Start', 'Cancel', 'Start');

        switch selection
            case 'Start'
```

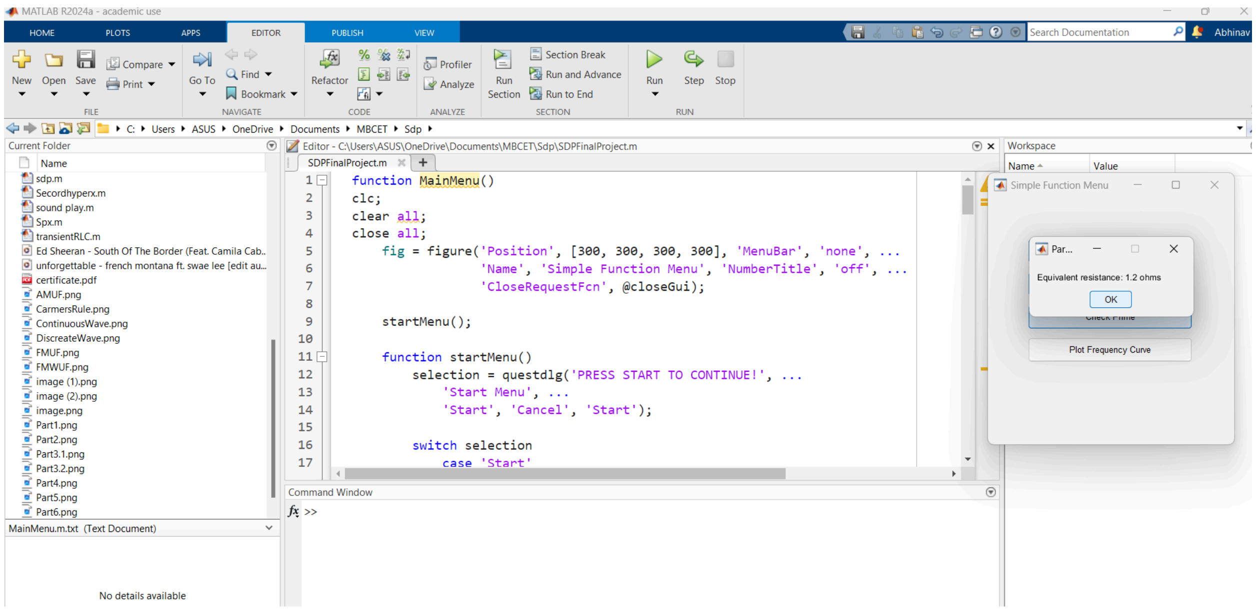
```
The first ten terms of the geometric progression are:
3   6   12   24   48   96   192   384   768   1536
```

Phase 4 :


Editor - C:\Users\ASUS\OneDrive\Documents\MBCET\Sdp\SDPFinalProject.m

```
        a = str2double(answer{1});
     2double(answer{2});
      = generateGP(a, r);
     tr = sprintf('The first ten terms of the geometric progression
     resultStr, 'Geometric Progression Result');


     elResistanceCallback(~, ~)
     Enter resistance R1 (ohms):', 'Enter resistance R2 (ohms):'};
     'Parallel Resistance Input';
     5];
     {'2', '3'};
     putdlg(prompt, dlgtitle, dims, definput);

     (answer)
```

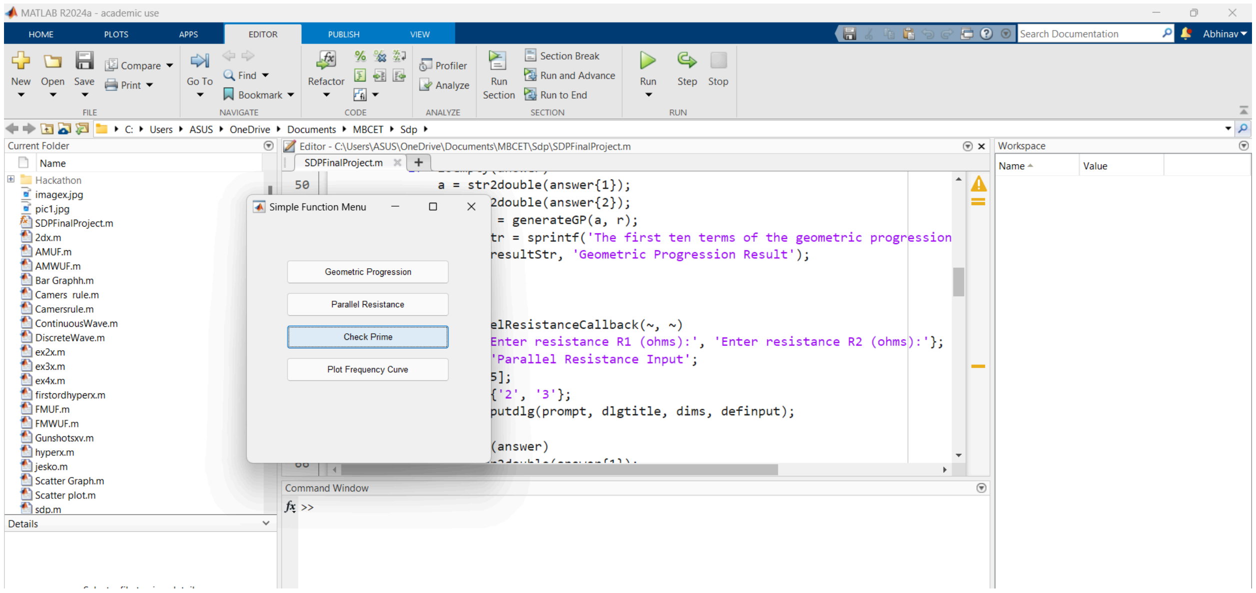
Output :


Editor - C:\Users\ASUS\OneDrive\Documents\MBCET\Sdp\SDPFinalProject.m

```
function MainMenu()
clc;
clear all;
close all;
    fig = figure('Position', [300, 300, 300, 300], 'MenuBar', 'none', ...
                 'Name', 'Simple Function Menu', 'NumberTitle', 'off', ...
                 'CloseRequestFcn', @closeGui);

    startMenu();

    function startMenu()
        selection = questdlg('PRESS START TO CONTINUE!', ...
            'Start Menu', ...
            'Start', 'Cancel', 'Start');

        switch selection
            case 'Start'
```

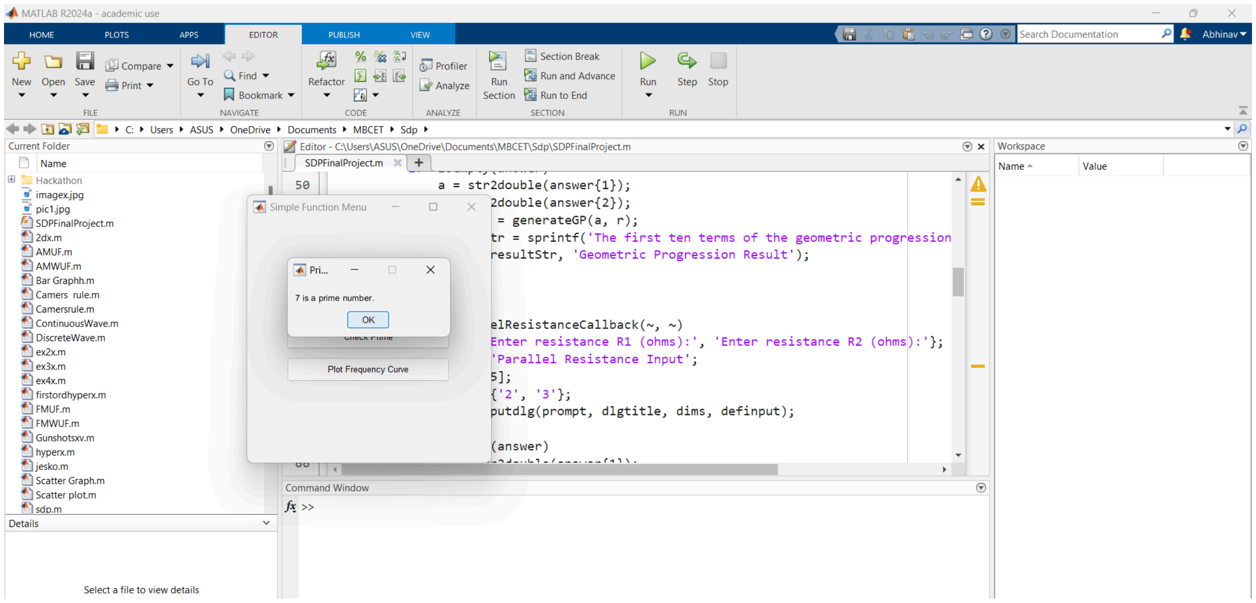
```
Equivalent resistance: 1.2 ohms
```

Phase 4:



```
    a = str2double(answer{1});
2double(answer{2});
= generateGP(a, r);
tr = sprintf('The first ten terms of the geometric progression
resultStr, 'Geometric Progression Result');


elResistanceCallback(~, ~)
Enter resistance R1 (ohms):', 'Enter resistance R2 (ohms):'};
'Parallel Resistance Input';
5];
{'2', '3'};
putdlg(prompt, dlgtitle, dims, definput);

(answer)
```

Output


MATLAB R2024a - academic use
Editor - C:\Users\ASUS\OneDrive\Documents\MBCET\Sdp\SDPFinalProject.m
Simple Function Menu
Pri...
OK
Plot Frequency Curve
Command Window
Workspace
Select a file to view details

```
7 is a prime number.
```

```
    a = str2double(answer{1});
2double(answer{2});
= generateGP(a, r);
tr = sprintf('The first ten terms of the geometric progression
resultStr, 'Geometric Progression Result');


elResistanceCallback(~, ~)
Enter resistance R1 (ohms):', 'Enter resistance R2 (ohms):'};
'Parallel Resistance Input';
5];
{'2', '3'};
putdlg(prompt, dlgtitle, dims, definput);

(answer)
```