

## Program\_07\_2

### Requirements

Given the following dataset, create a script that satisfies the following requirements.

#### % Dataset

```
x = [10 20 30 100 200 300 500 700 1000];  
y = [18 38 49 82 102 113 127 137 146];
```

- Determine the functional form of an (x,y) dataset by creating plots and figures
- Create a Figure 1 window
- Create a Figure 2 window

Figure 1:

- Have four plots, `plot(x,y)`, `loglog(x,y)`, `semilogx(x,y)`, and `semilogy(x,y)`
- Provide a title, and axis labels for all four plots using your previously generated function **labelPlot**

Figure 2:

- Use **polyfit()** along with the correct linear model to find the **m** and **b** coefficients.
- Provide the functional form, along with numerical values of the best fit coefficients in a text object on the plot.
- Use the fit equation to create a linear plot of the data and the fit equation plotted with at least 100 points.
- Format the plot with blue asterisk markers, and a solid red line.
- Add a legend indicating the functional form of the fit line in the lower right
- Provide a title, and axis labels for the plot using your previously generated function **labelPlot**

### Program

In the code block below, create your program, editing the existing text as necessary.

**Note:** If you are using Octave then you will need to create a separate script file, save that separate file as the name **Program\_07\_02**. It will not conflict with this file of the same name since the extension will be different.

#### Tips:

- The plot (of the four subplots) showing a linear representation of the data will indicate the functional form of the data set, refer to the **Tutorial\_07\_5** for more detail.
- Remember, you can use manual calculations or the **polyval()** function
- You cannot **fprintf** to a title but you want the variable data to be displayed in text. Recall that the **sprintf** function allows you to create a string of text much like `fprintf` but you can store the result in a variable.

```
% Filename: Program_07_2  
% Author:  
% Assisted by:  
  
% Program Description:
```

## Example Output

When running your program from the **Command Window**, your output should match the following.



