Numerical Analysis and Programming

Lab Worksheet #6

1. *List Comprehension* List comprehensions provide a concise way to create lists. Each list comprehension consists of an expression followed by a `for` clause, then zero or more `for` or `if` clauses. The result will be a list resulting from evaluating the expression in the context of the `for` and `if` clauses which follow it. If the expression would evaluate to a tuple, it must be parenthesized. Here are some examples:

```
>>> vec = [2, 4, 6]
>>> [3*x for x in vec]
[6, 12, 18]
>>> [3*x for x in vec if x > 3]
[12, 18]
>>> [3*x for x in vec if x < 2]
[]
>>> [[x,x**2] for x in vec]
[[2, 4], [4, 16], [6, 36]]
```

   Using list comprehension to generate the transpose matrix and trace of the matrix,

```
>>> M=[[1,2,3],
   [4,5,6],
   [7,8,9]]
```

2. *Data Analysis* The data file `Data.in` contains the exam scores of a class, in the comma-separated-value (csv) format. Each row corresponds to the scores of a student, and each column is the score that the student got for each problem. Write a short program to perform the following tasks:

3. Find the maximum, minimum, average, median, and standard deviation of the exam scores, using `numpy` supplied functions.

4. Use the function `hist` in *matplotlib* to plot the histogram of the score distribution with 10-point bin intervals.