**ENR 261 Spring 2023 Chapter 11 Homework**

**General Instructions:**

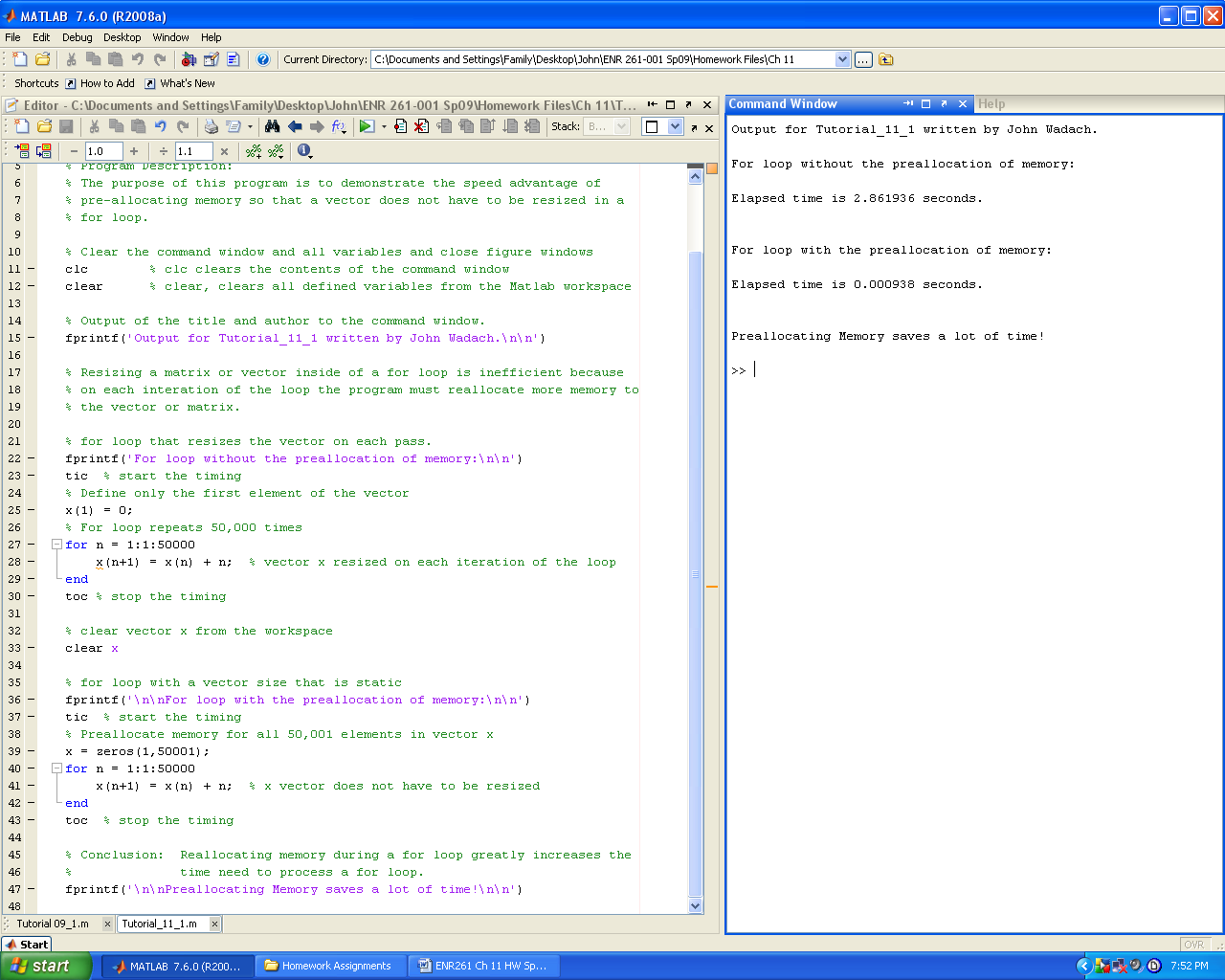
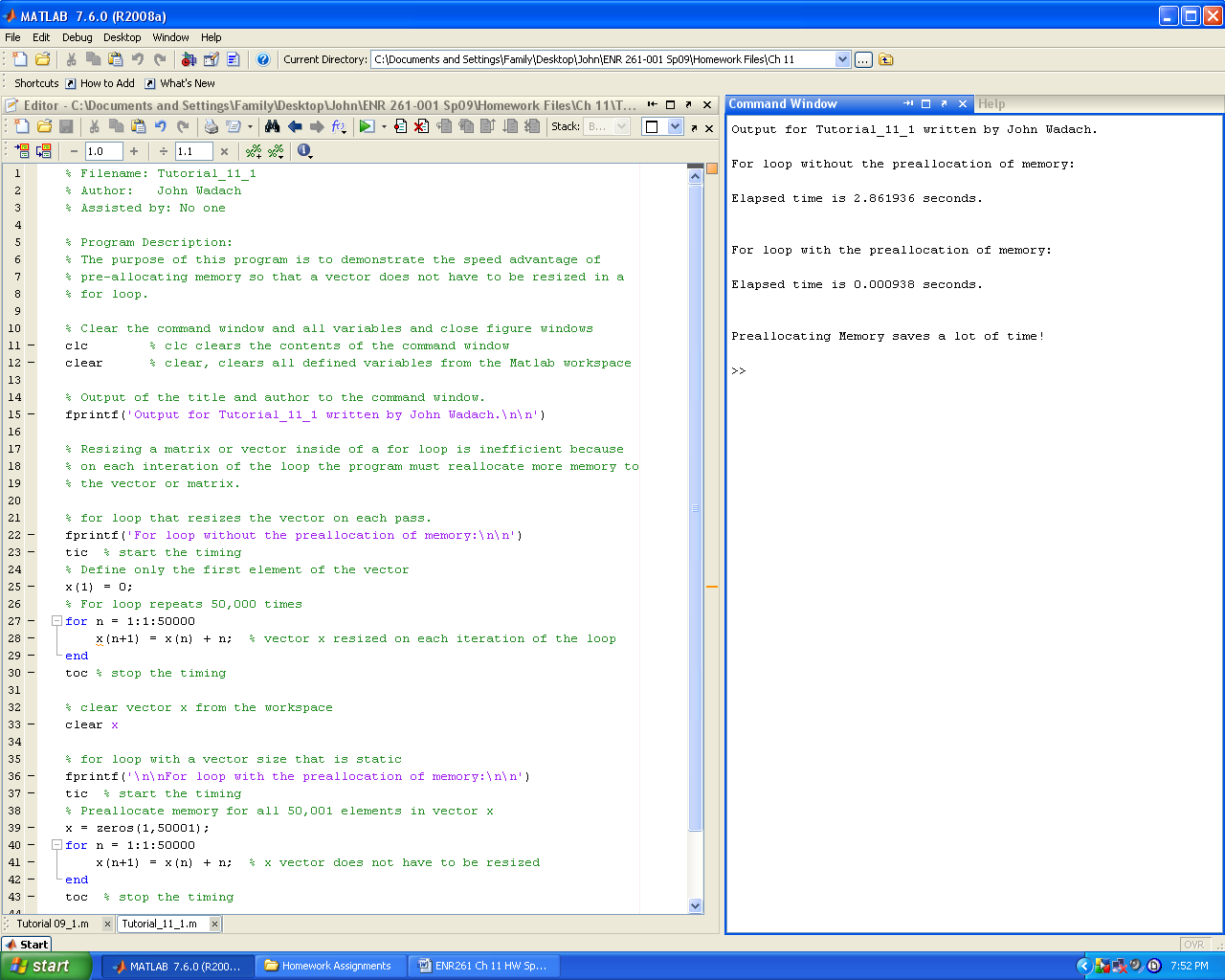
Save your all your Matlab files for this chapter in the folder named **Ch11** located inside your local repository on your USB Memory Stick. When finished be sure to add, commit, and push your changes to your remote repository on GitHub.

**Assigned Exercises**

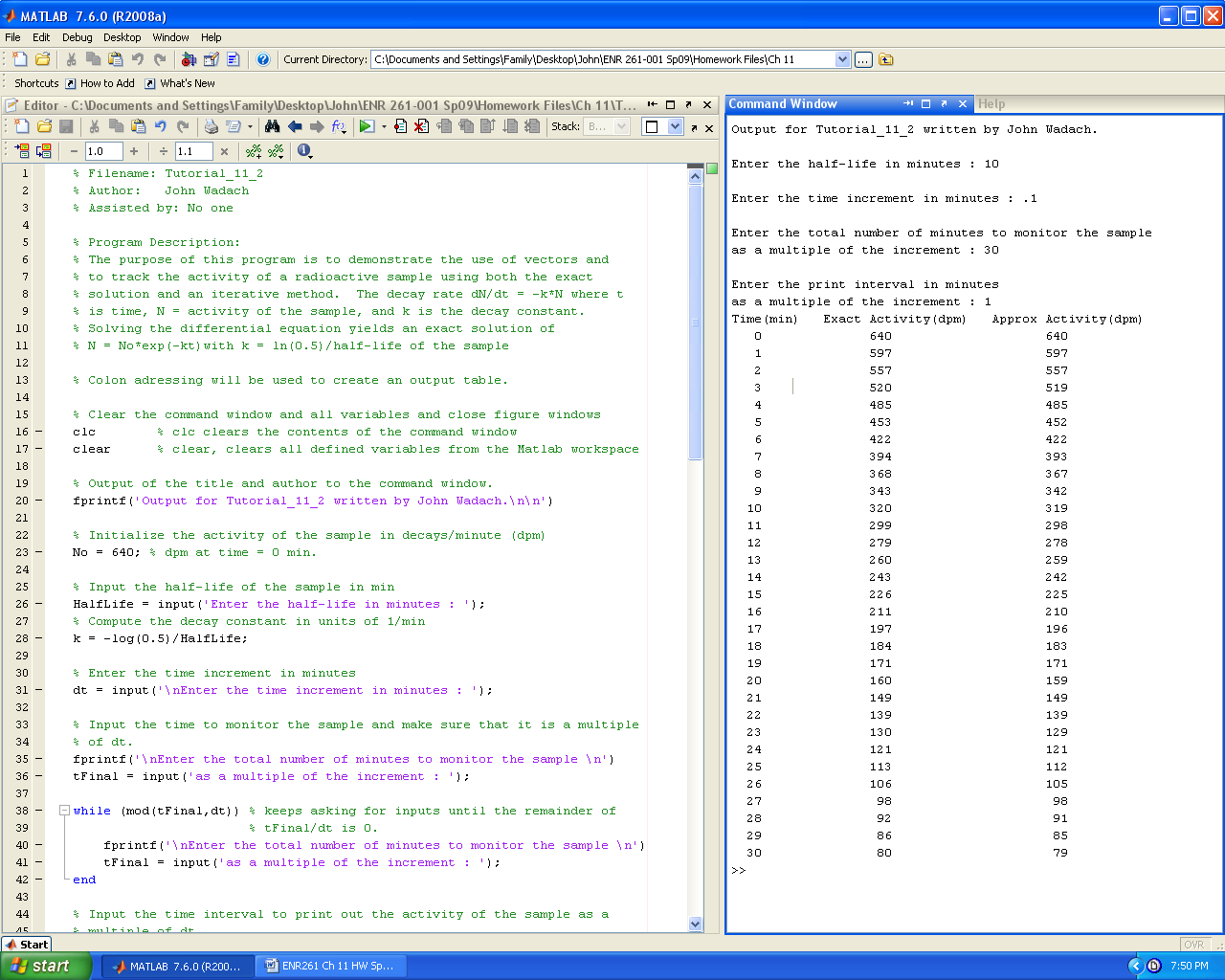
1. Recreate all of the following script files and be sure to save them in your local repository on your USB memory stick, commit the changes and push them to GitHub.

2. Use the required file names for each script file.

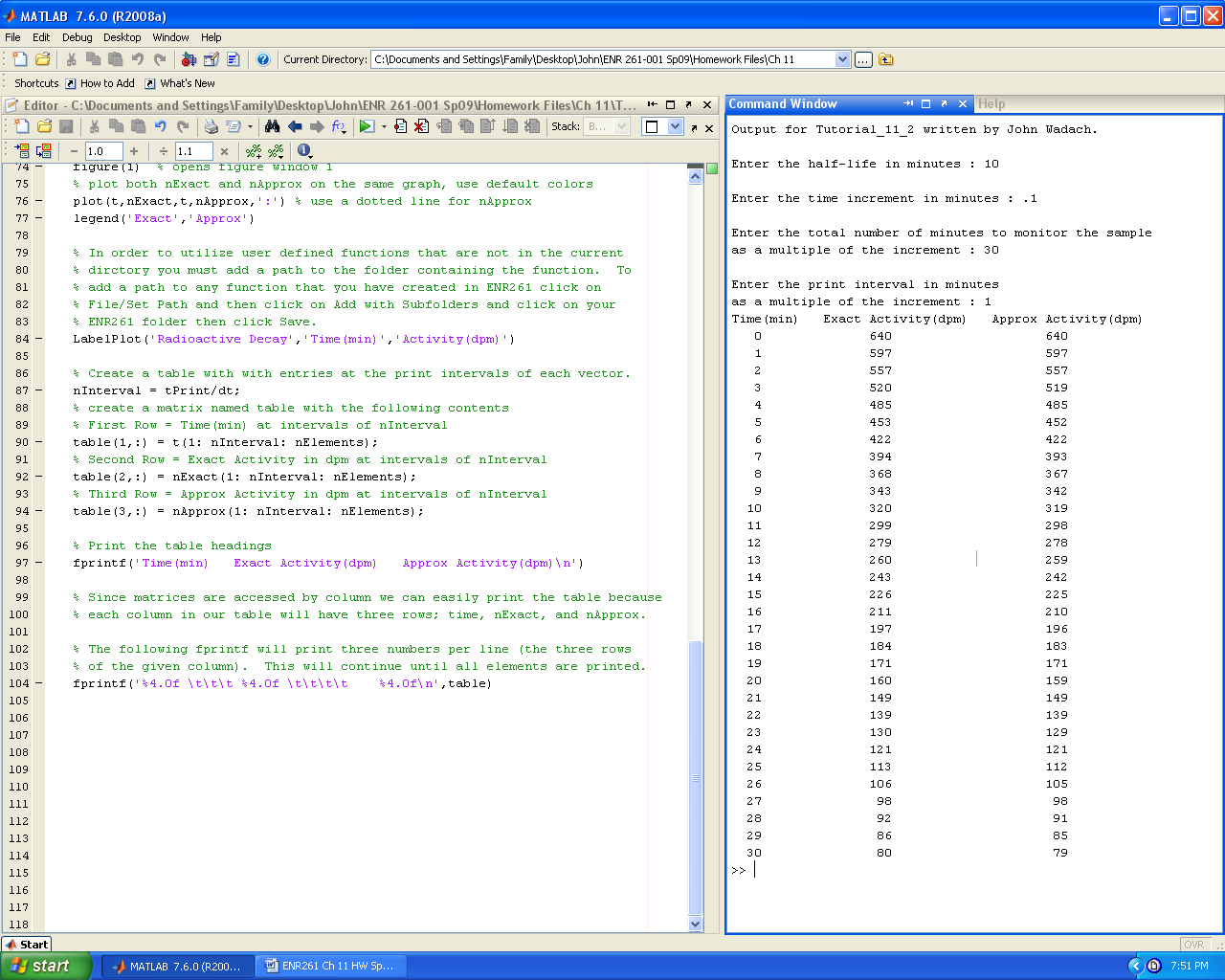
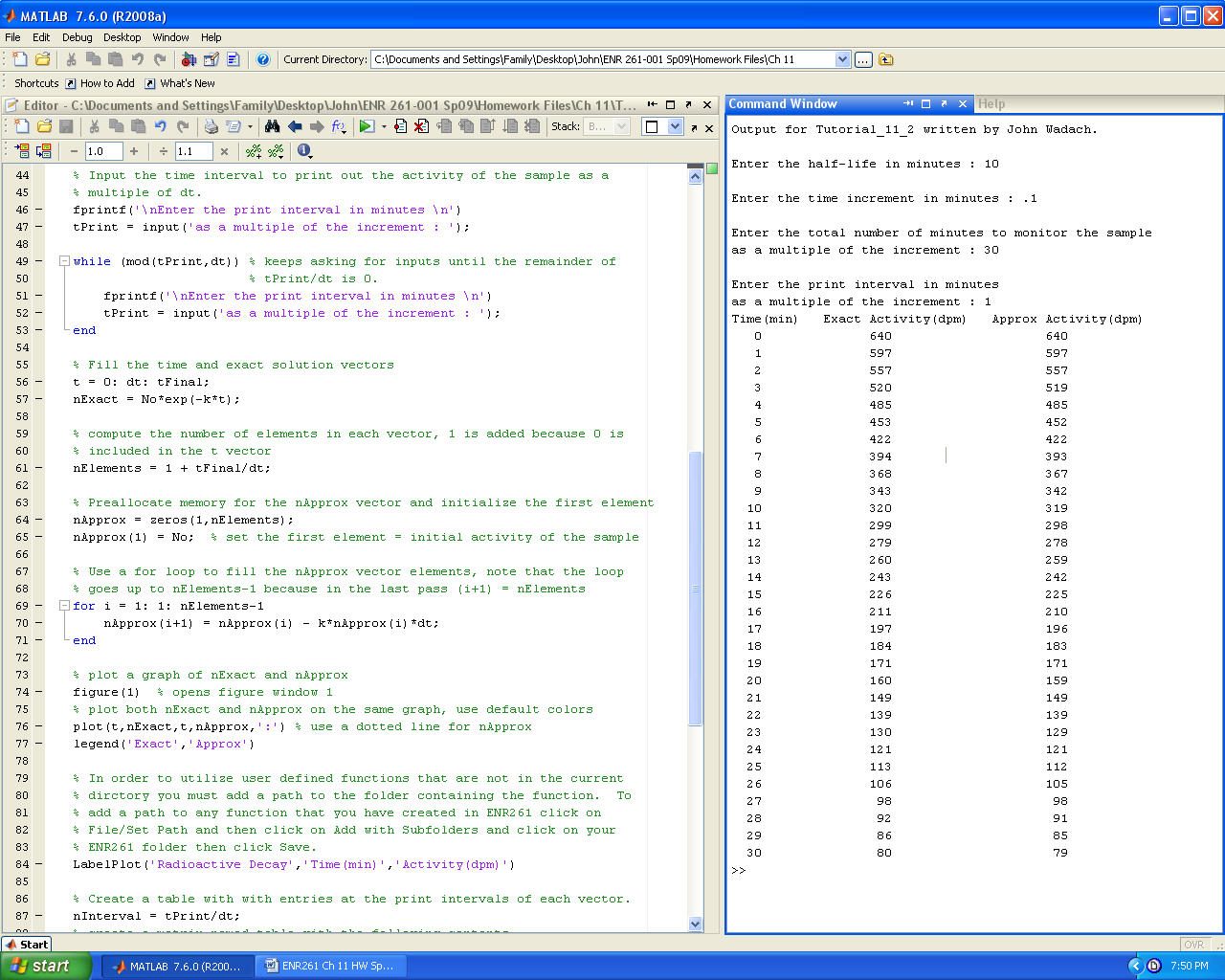
Required File Name: **Tutorial\_11\_1.m**

****

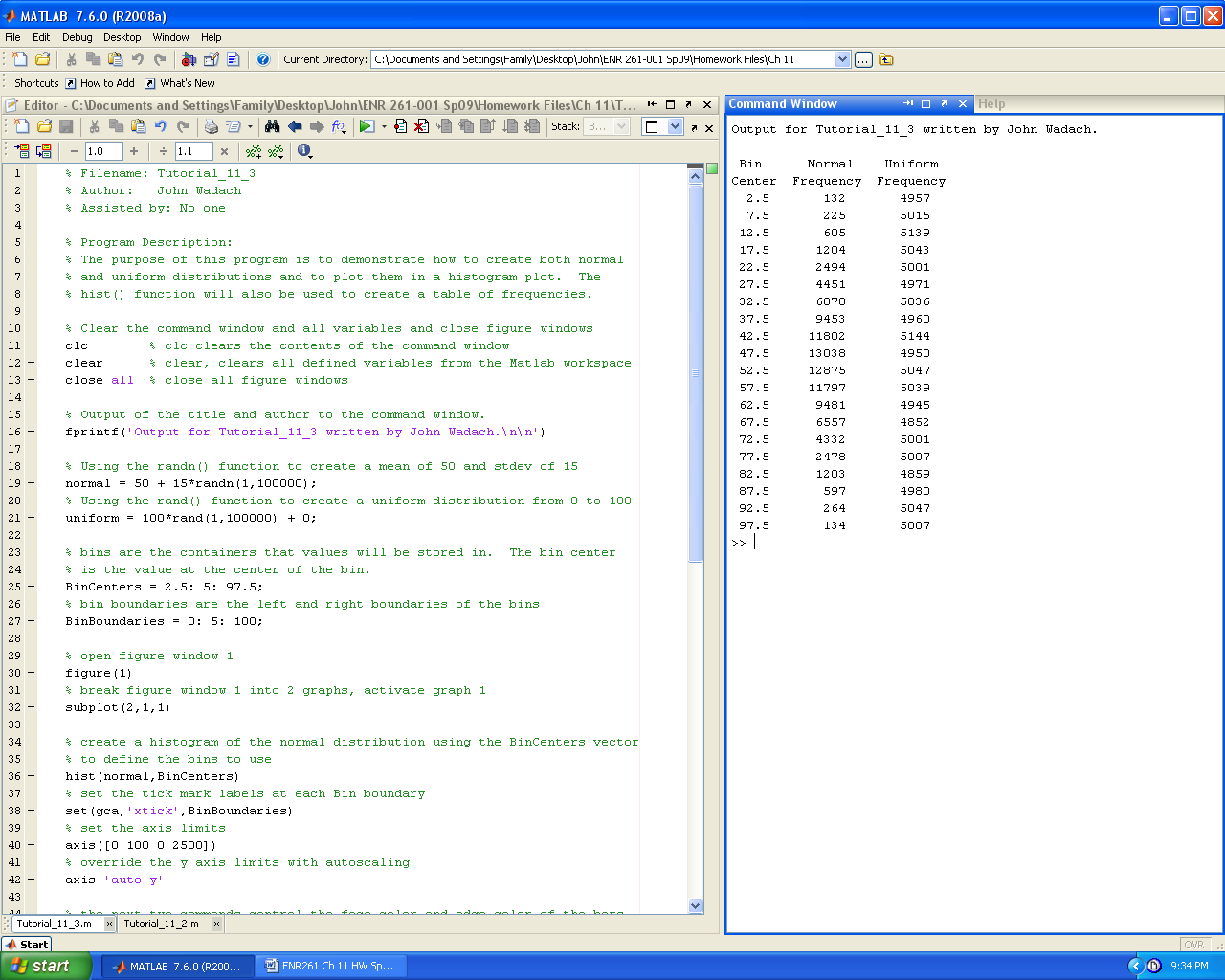
Required File Name: **Tutorial\_11\_2.m**

****

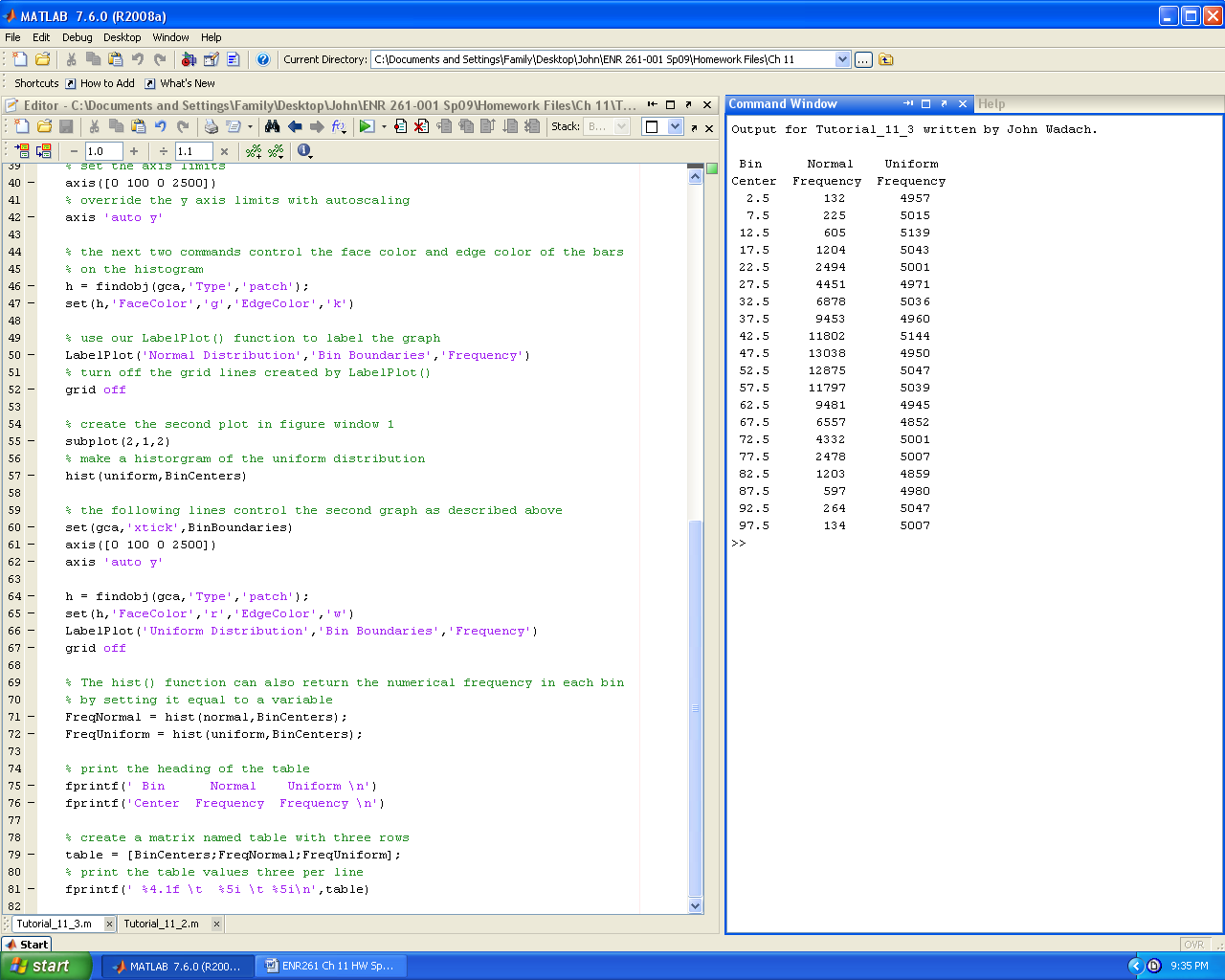
**Continued on the Next Page**

****

Required File Name: **Tutorial\_11\_3.m**

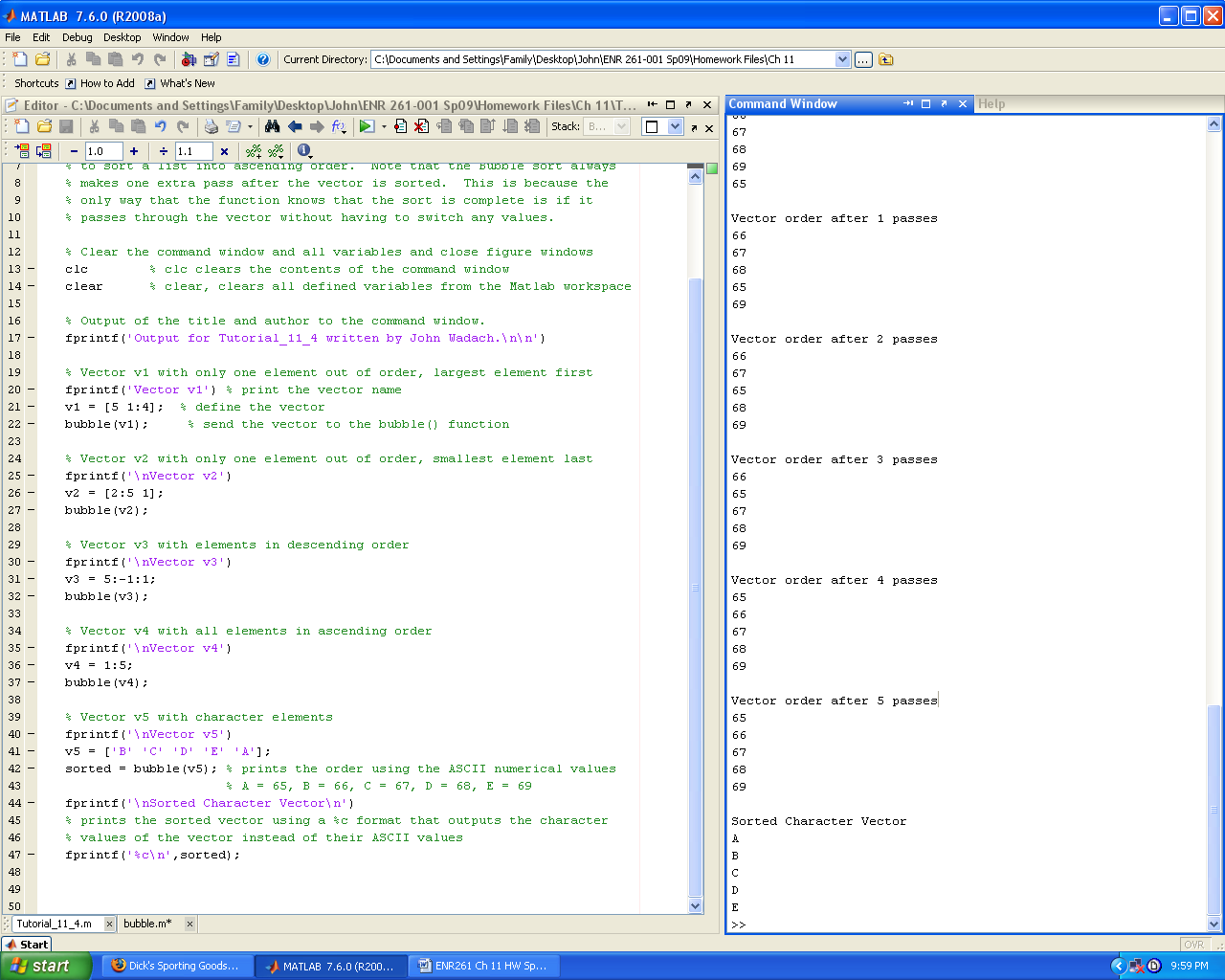
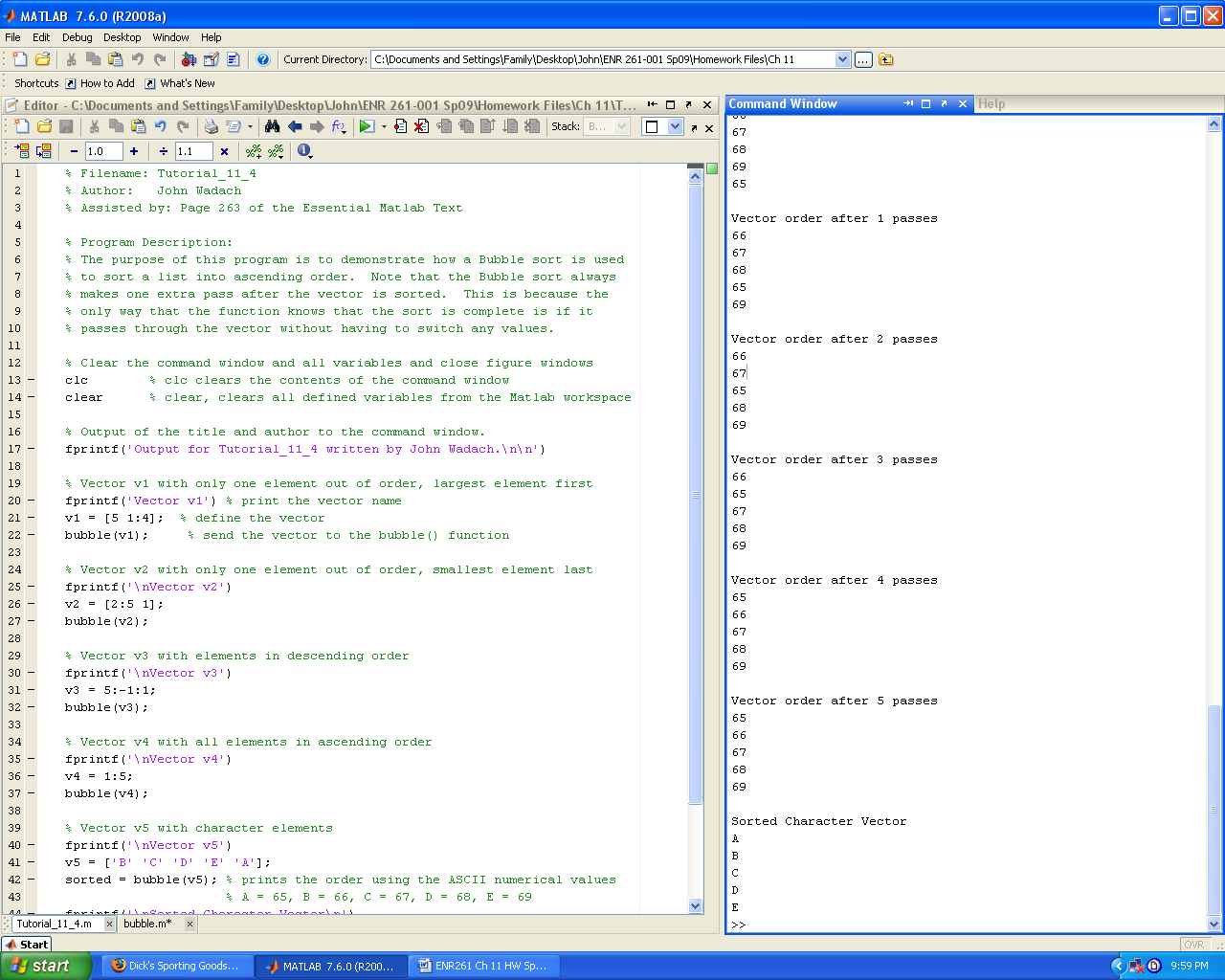
****

**Continued on the Next Page**

****

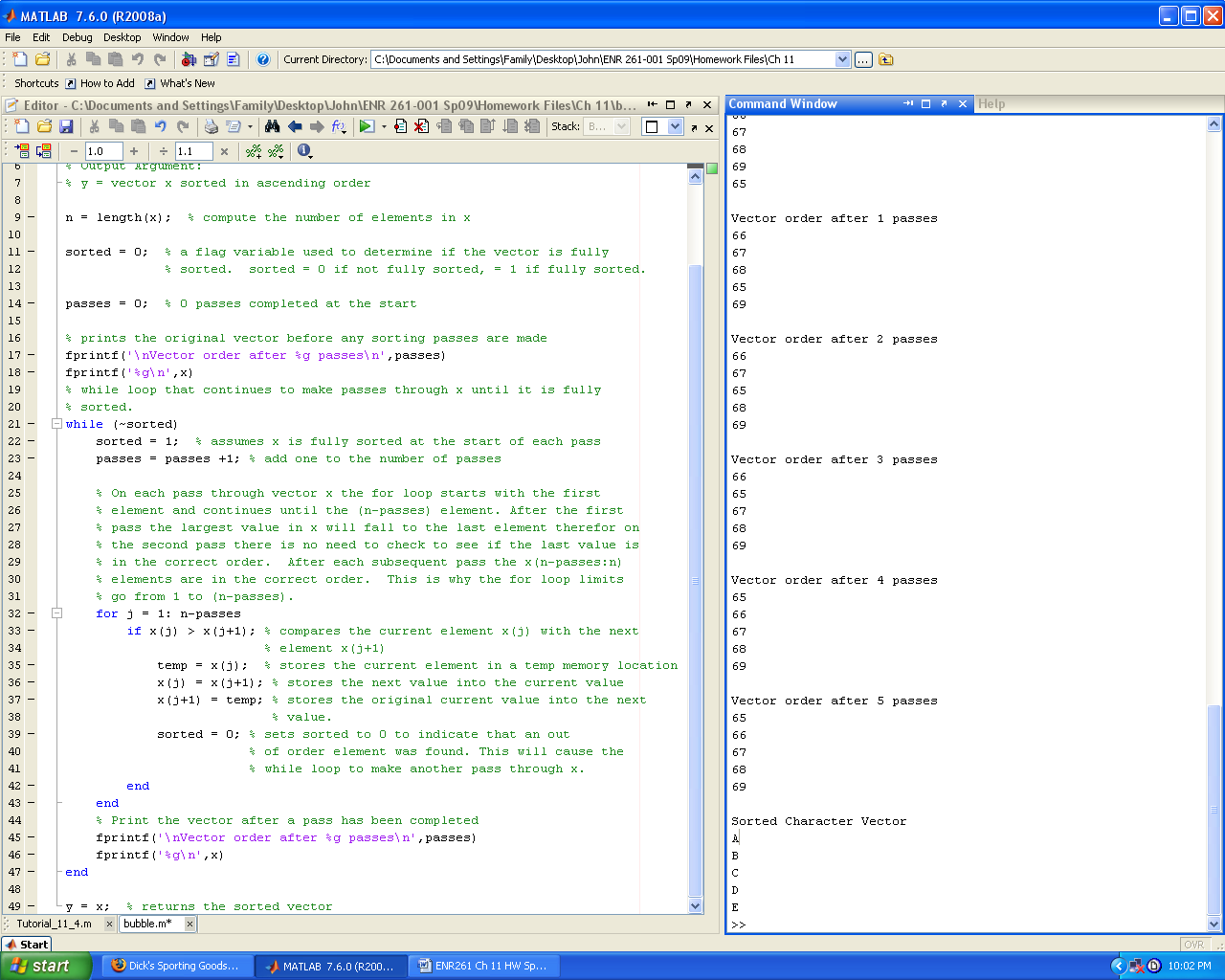
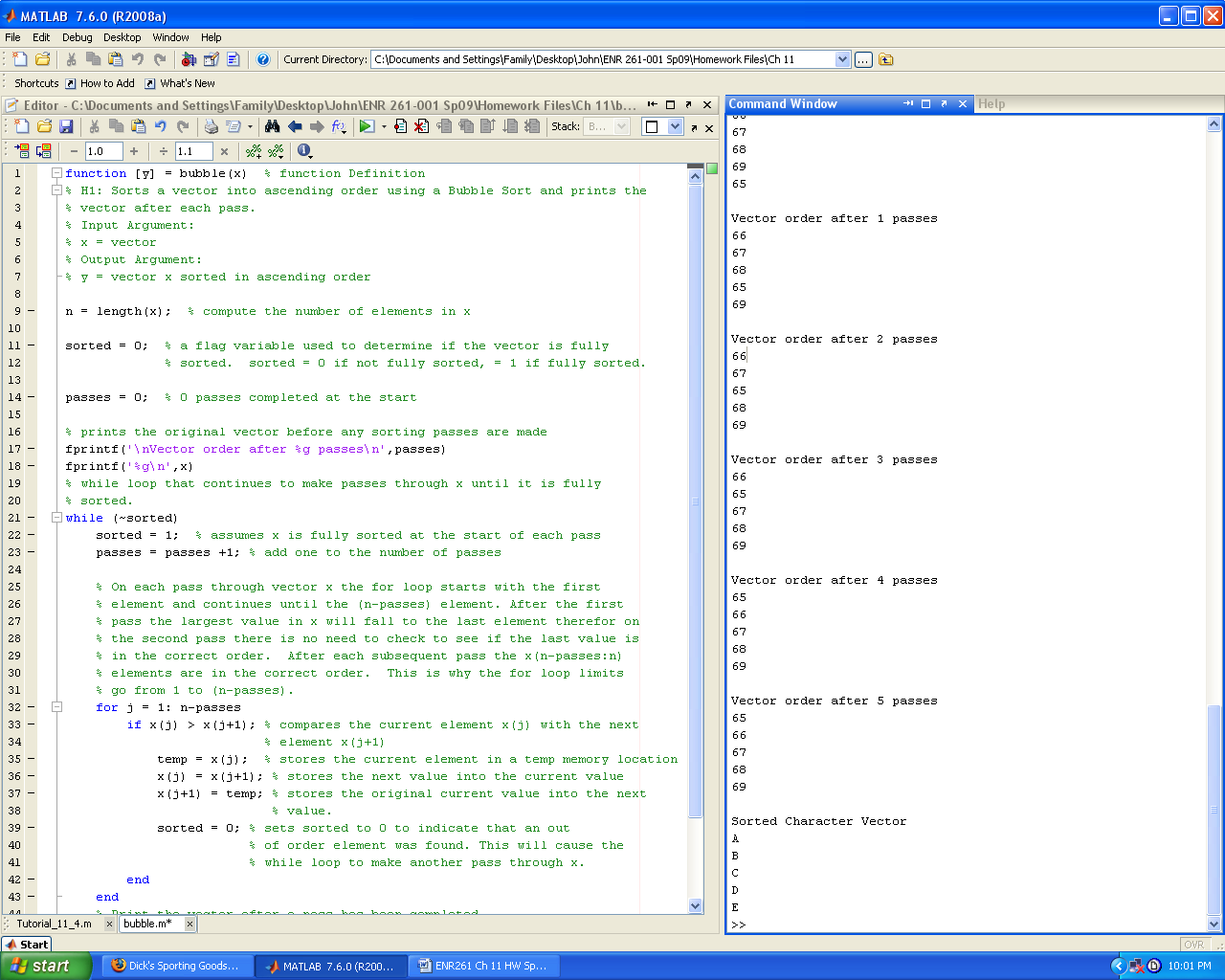
Required File Name: **Tutorial\_11\_4.m**

**You will also have to create function bubble() in order for this program to work.**

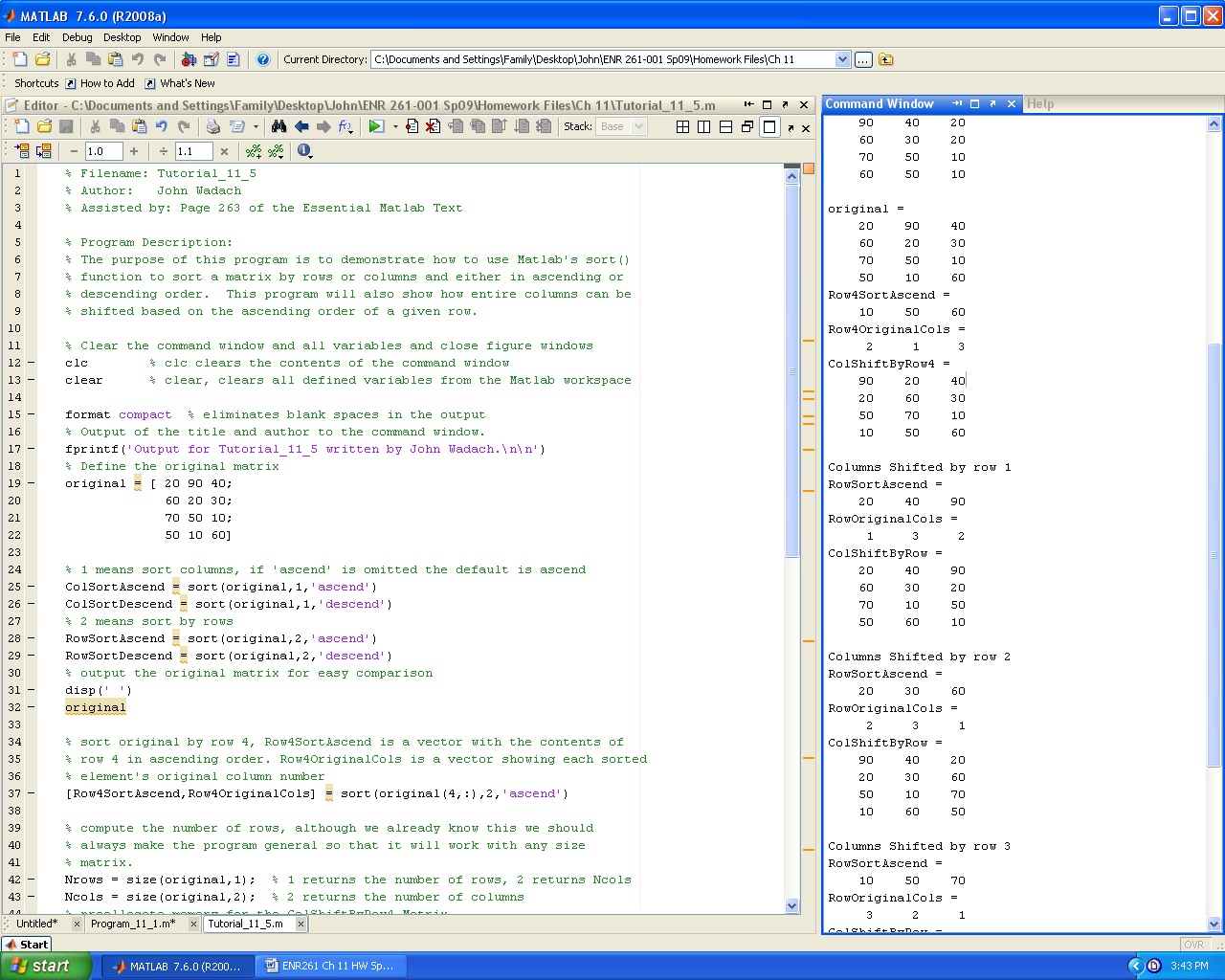
****

**Continued on the Next Page**

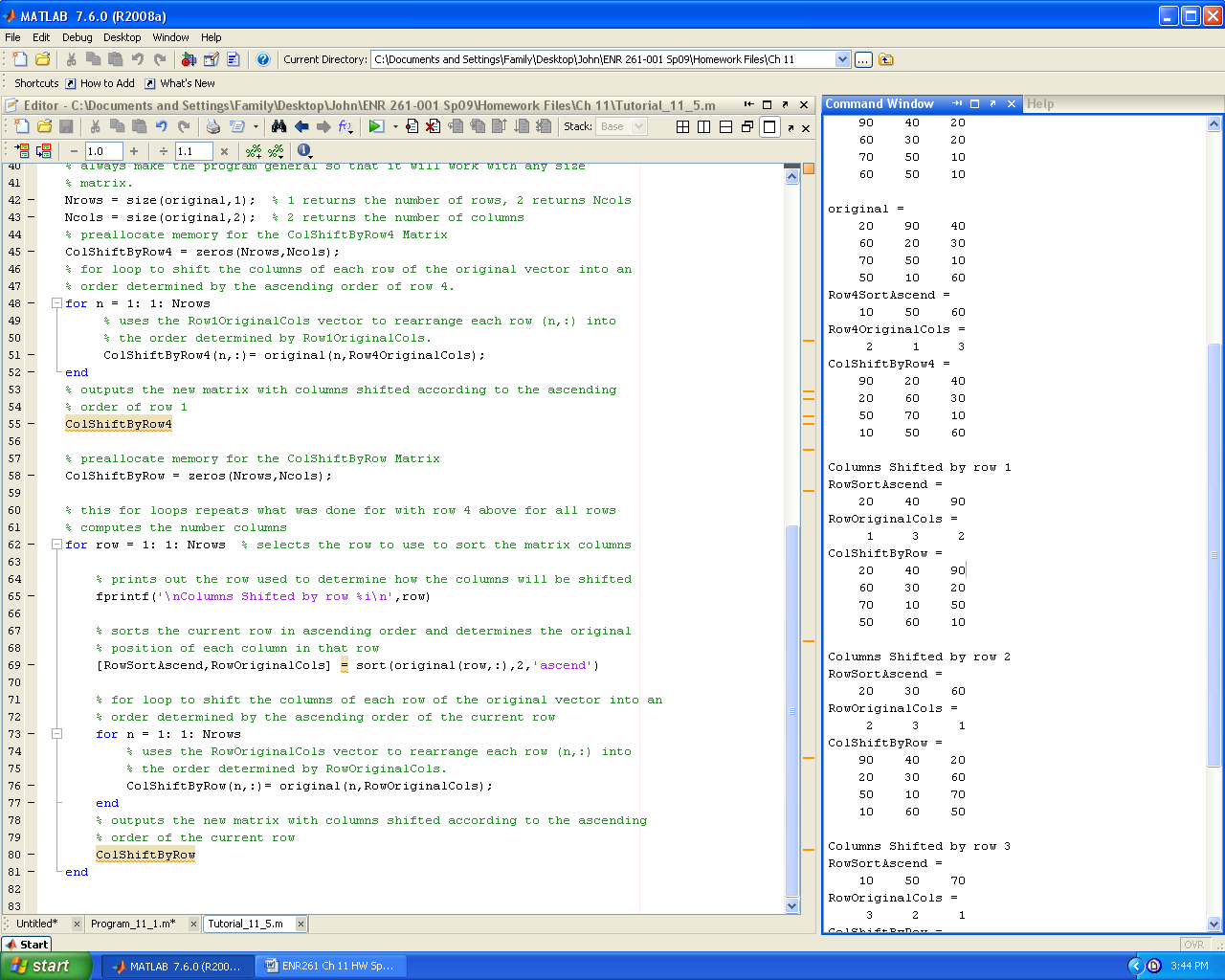
Required File Name: **bubble.m**

****

Required File Name: **Tutorial\_11\_5.m**

****

**Continued on the Next Page**

****

Required File Name: **Program\_11\_1.m**

% Program Description:

% The purpose of this program is to read in design rating data from the

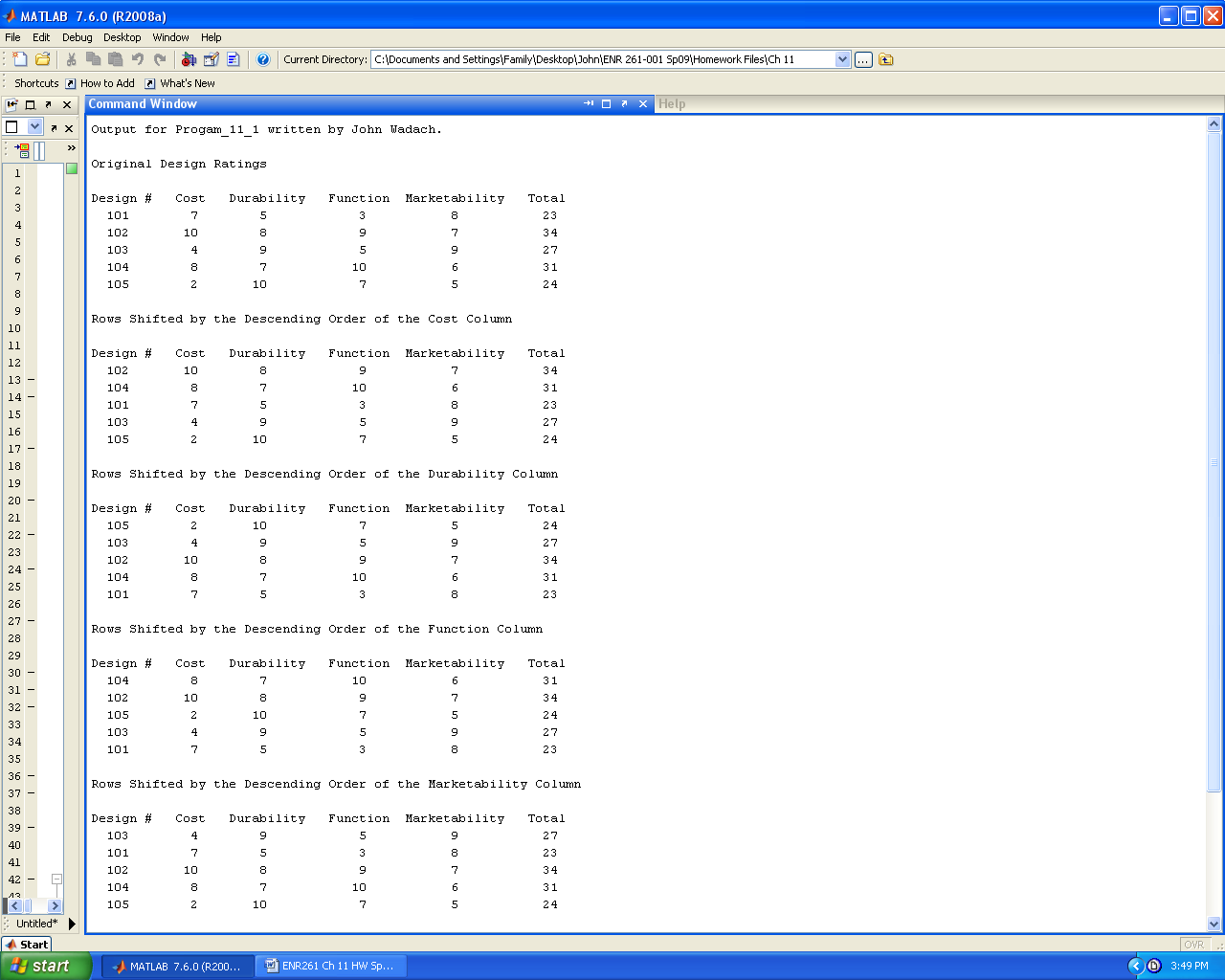
% file named Data\_11\_1.txt.

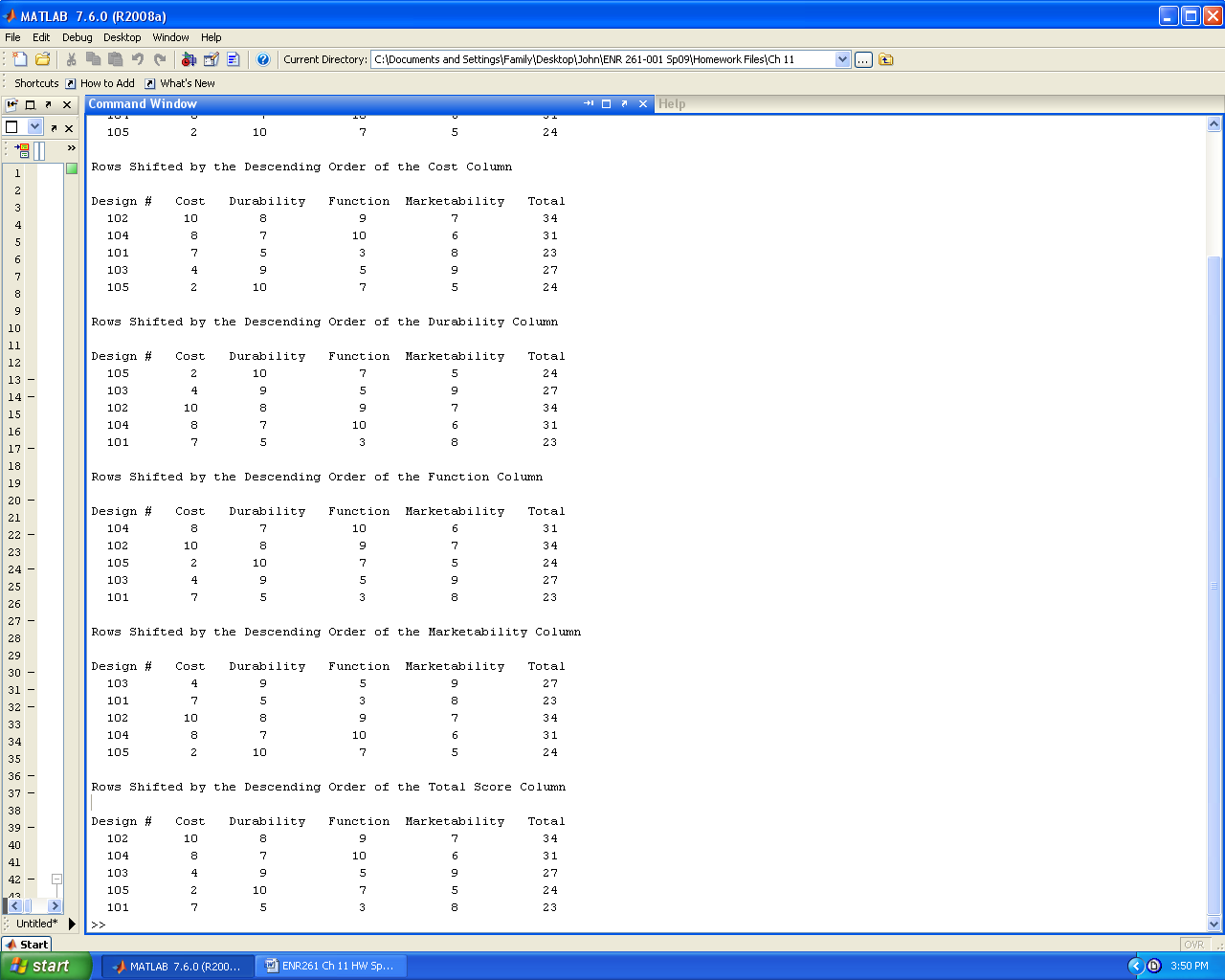
% The data is output in table form in its original form then sorted in

% descending order according to the cost, durability, function,

% marketability, and total score columns.

A sample of the output is shown below.





Required File Names: **Program\_11\_2.m , DateSort.m, Over21.m**

**% Program Description:**

% The purpose of this program is to read in birthdates from the

% file named BirthDates.txt.

% The data is output in table form in its original order then sorted in

% order of ascending age using a user defined function named DateSort().

% The user defined function Over21() is used to determine ages above and

% below 21 as of the date entered by the program user.

% Two tables of sorted dates are then printed out.

% The first showing birth dates < 21 years old as of a date

% input by the user and the second for those older than 21.

**function [y] = DateSort(x) % function Definition**

% H1: Sorts a matrix of dates into descending order using a Bubble Sort

% Input Argument:

% x = matrix of dates with the following columns: 4 digit year,

% 2 digit month, 2 digit day

% Output Argument:

% y = matrix with the x in descending order (youngest to oldest)

**function [y] = Over21(x,year,month,day) % function Definition**

% H1: Determines if an age is over 21 years old as of a given year,month,day

% Input Arguments:

% x = matrix of dates with the following columns: 4 digit year,

% 2 digit month, 2 digit day

% year is the current year

% month is the current month

% day is the current day

% Output Argument:

% y = logical vector with a 1's in positions with ages >= 21 and 0's in

% positions with ages < 21.

You should use importdata() to import the birthdate information, the data will be imported as a struct to take some time to figure out how to extract the numerical data.

A sample of the program is shown below and on the next page. **Test your program with several different current dates.**

Output for Progam\_11\_2 written by Geoff Berl.

Original Dates from BirthDates.txt

Year Month Day

1997 04 14

2018 07 04

2008 08 21

1988 04 14

1980 11 30

1988 03 05

1988 02 29

1997 04 12

1988 04 15

1988 12 01

1990 04 15

1997 04 13

1960 07 07

2002 02 02

2012 02 09

1982 10 30

1993 09 25

2003 08 14

1978 06 17

1988 04 16

1997 04 15

1980 11 29

1982 08 17

1990 05 01

2002 11 11

2012 12 25

1997 04 16

1990 01 01

1989 03 22

Birth Dates Sorted from Youngest to Oldest

Year Month Day

2018 07 04

2012 12 25

2012 02 09

2008 08 21

2003 08 14

2002 11 11

2002 02 02

1997 04 16

1997 04 15

1997 04 14

1997 04 13

1997 04 12

1993 09 25

1990 05 01

1990 04 15

1990 01 01

1989 03 22

1988 12 01

1988 04 16

1988 04 15

1988 04 14

1988 03 05

1988 02 29

1982 10 30

1982 08 17

1980 11 30

1980 11 29

1978 06 17

1960 07 07

Input the current 4 digit year: 2018

Input the current 2 digit month: 04

Input the current 2 digit day: 12

Birth Dates with Ages Less Than 21 Years

Year Month Day

2018 07 04

2012 12 25

2012 02 09

2008 08 21

2003 08 14

2002 11 11

2002 02 02

1997 04 16

1997 04 15

1997 04 14

1997 04 13

Birth Dates with Ages Greater Than 21 Years

Year Month Day

1997 04 12

1993 09 25

1990 05 01

1990 04 15

1990 01 01

1989 03 22

1988 12 01

1988 04 16

1988 04 15

1988 04 14

1988 03 05

1988 02 29

1982 10 30

1982 08 17

1980 11 30

1980 11 29

1978 06 17

1960 07 07