1. You should notice that a folder was created by the unzipping process called "Pep7v0".  If you inspect the contents of that folder, you'll see a file called "Pep7.exe" in the folder.  This is the filename you will use to run the Pep/7 simulator this semester.  Launch this program now, and click on the Maximize button in the upper-right corner of the window border to run the program in full-screen mode.  
     
   Choose "File/New" from the menu bar, and enter the following code EXACTLY AS IT APPEARS HERE (except that you may use lower-case letters if you choose).  The last two characters MUST be lower-case z's.  Be careful NOT to include any spaces or blank lines before the first character (E), and to use EXACTLY one space between each two-character hex code.
2. E1 00 13
3. E1 00 14
4. E1 00 15
5. E1 00 16
6. E1 00 17
7. E1 00 18
8. 00
9. 48 65 6C 6C 6F 21 zz

After entering this machine language program, choose "File/Save As" from the menu bar, save the file with the name ***prog1lab1***.

1. Now that we have entered a machine language program into the Pep/7 simulator, how do we execute the program?  To do this, open the "Pep7" menu from the menu bar.  When working in machine language, there are two steps to running a program; we must first LOAD the program into memory, and then EXECUTE the program.  The "Assemble" option of the menu won't be used until we start working in Pep/7 Assembly language.    
     
   Now, choose "Load" from the Pep7 menu -- if you didn't make any mistakes in entering your program, nothing much happens on the screen; if you did make some typing errors, the program won't load, and the simulator displays an error dialog box.  After ensuring that we have no typing errors, our program is now ready to run, so open the Pep7 menu again, and notice that the "Execute" option is now selectable (it wasn't before we loaded our program).  Choose the "Execute" option, and behold the output of your first Pep/7 program!  We'll pause now for just a second so that you can catch your breath!!   
     
   Open the Pep7 menu again, and notice in the second section of the menu there is an option called "Load/Execute", which you can use to perform both of the required operations with one mouse click.  This is the option you will most often use to run programs.
2. The Pep/7 simulator requires that each byte of a machine language program be entered with no address information, with exactly one space between each two hex-digit byte code, and with no extra spaces at the end of any line.  The example program you just ran would be written this way, using the program listing style we've seen in class and in chapter 4 of the Warford text:
3. Addr Machine Language
4. ---- ----------------
5. 0000 E10013 ; character output
6. 0003 E10014 ; character output
7. 0006 E10015 ; more character output
8. 0009 E10016 ; still more character output
9. 000C E10017 ; even more character output
10. 000F E10018 ; and again!!
11. 0012 00 ; stop
12. 0013 48 ; ASCII code for 'H'
13. 0014 65 ; ASCII code for 'e'
14. 0015 6C ; 'l' -- that's a lower-case L
15. 0016 6C ; 'l'
16. 0017 6F ; 'o'
17. 0018 21 ; '!'

When entering machine language code into the simulator, we don't enter any address information, and we indicate the end of all byte-code entries in the file with the special terminator sequence 'zz', which MUST be lower-case.  Now, let's see if we can modify this program.

1. Can you change ***prog1lab1*** so that it prints HELLO! using all upper-case letters?  After making that change, modify the program so that the message HELLO THERE! is printed on screen, using all upper-case letters.  When you finish these two modifications, save the program in the **labx01**folder using the name ***prog2lab1***, and show the output of the program to your instructor.
2. Close the current Pep/7 program window, and choose "File/New" to open a blank editor window.  Enter the following machine language program into the editor window:
3. Addr Machine Language
4. ---- ----------------
5. 0000 E1002E
6. 0003 E1002F
7. 0006 E10030
8. 0009 E10031
9. 000C E10032
10. 000F D90033
11. 0012 D90034
12. 0015 D90035
13. 0018 E10036
14. 001B E10037
15. 001E E10038
16. 0021 E10033
17. 0024 E10034
18. 0027 E10035
19. 002A E10039
20. 002D 00
21. 002E 4E
22. 002F 61
23. 0030 6D
24. 0031 65
25. 0032 3F
26. 0033 00
27. 0034 00
28. 0035 00
29. 0036 48
30. 0037 69
31. 0038 20
32. 0039 21

After entering this machine language program, save the file using the name ***prog3lab1*** in your **labx01** folder.  
  
Notice that this program has some input instructions (D9 instrucution specifier).  Before running such a program, we need to tell Pep/7 the source of the input.  To do this, open the Pep7 menu, choose the "Execution Input" option, and select "Interactive Input from Keyboard".  Then run the program.  What happens?  Run the program again, and enter "Abraham" as the name.

1. Change the program as follows: immediately after the last input (D9) instruction, insert the following instruction:
2. E1 00 3D

and enter a new last byte to the program:

0D

Also, since we've added a new instruction to the middle the program, you'll need to edit each operand specifier (except for the 003D you just entered!) and add 3 to its address.  See what happens when you make that change and run the program.  Explain to your instructor what ASCII code 0D does when output by the Pep/7 simulator.

1. If any time remains in today's class, you may work on Assignment # 5