CSC231: Computer Organization and Assembly Language

Lab #6: Assembly Language Lab Tutorial

This tutorial provides an introduction to the 8088 Assembler and Tracer Toolkit which is available in the CS lab. This is an important lab because it will familiarize you with the tools you will require to complete the remaining assignments for this course.

First read Appendix C in your textbook to get an introduction to assembly language programming and the 8088 processor. ***Do not skip this step!*** Understanding this background material is important before you begin writing your own assembly language programs! As you work on your assembly language programs you may want to keep Figure C-4 handy as a reference summarizing the most important 8088 instructions.

Begin with the “Hello World” example in section C.8.1. Enter the program below using **arc**

```
! Simple "hello world" program
     _EXIT = 1
     _PRINTF = 127

.SECT .TEXT
     PUSH hw
     PUSH _PRINTF
     SYS
     ADD SP,4
     PUSH _EXIT
     SYS

.SECT .DATA
hw:  .ASCIZ "Hello World\n"
```

Once you have entered the program, you can run it line by line by either:

1. clicking the “Run One Line” button at the bottom of the software
2. clicking the “Edit” menu, then the “Run One Line” menu item
3. pressing “Enter” at any time (as long as the input text field is not focused)

The software will allow you to run, debug, or single step through the program . **Arc** differs from the t88 software presented in the text.

Change the .ASCIZ declaration to a .ASCII declaration, what changes? What if you keep the declaration as .ASCII and add the following line:

```
extra:     .ASCII "Ignore me, I should never show up"
```

What is happening?

Try modifying the program by adding the two underlined lines:

```
! Simple "hello world" program
     _EXIT = 1
     _PRINTF = 127

.SECT .TEXT
start:
     PUSH hw
     PUSH _PRINTF
     SYS
     ADD SP,4
     PUSH _EXIT
     SYS
     JMP start

.SECT .DATA
hw:  .ASCIZ "Hello World\n"
```

The **start:** line defines what is called a *label*. Let's pretend we couldn't puzzle out what JMP does. Arc comes with documentation of all instructions. If you navigate to the bottom right of the expanded viewport you will see the *Instruction Help* tab. This will include documentation on the JMP command.

Continue by doing the following the examples:

- Make a program where the computer prints all the numbers from 1 to 10 (hint, do it once hardcoded, then try to use a loop)
- Adapt the above program to now print out the A,B,C's

Note that source code for these examples can be found on the CD-ROM that accompanies the textbook. Note that you can debug your code using the 8088 tracer toolkit (t88). Once your program is debugged and tested, you can run your program using the 8088 simulator (s88).