NAME

# CIS40/CNET220 EXERCISE 2A
# READING AND WRITING USING ASSEMBLY LANGUAGE

The following are examples for reference, but are not the actual exercise.
(exercise starts at PROBLEM STATEMENT)
**Example** of loading DS segment register with segment value of 4000 (note your segment is different)
Note it is a two step process since immediate values cannot be read into segment register directly

```
MOV AX, 4000
MOV DS, AX
```

**Example** of an *immediate* move loading AX with 1000 (note your immediate value is different)

```
MOV AX, 1000
```

**Example** of using *direct* mode for a write operation:

```
MOV [100] , AX
```

This will write 1000 hex stored in AX to offset address 100 hex of the current segment (which is 4000 hex)

**Example** of using *direct* mode for a read operation:

```
MOV  BX, [100]
```

This will read 1000 hex stored in offset address 100 hex of the current segment (which is 4000 hex) into the BX register

## PROBLEM STATEMENT

Summary: At the *default* CS and IP address assemble a program that will write ABCDh to 3000:2000 then read it back, then write it to 2000:1000. Note for the default assembly address enter "a" only in debug

1. Go into the assembler (type "a" at the debug command line) assemble at the default address (should be offset 100 in the current code segment).
2. Turn the following steps into assembly language and enter the instructions:
3. Read ABCD to the **AX** register using an *immediate* mode move statement
4. Write the **AX** register (with contents ABCD) to address **3000:2000** using *direct* mode
5. Read address **3000:2000** to register **BX** using *direct* mode
6. Write the contents of BX register to address **2000:1000** using *direct* mode
7. Note the address *after* the last instruction entered in the assembler and record it__________________
8. Select the enter key one last time to exit the assembler
9. Use the Unassemble command (u 100) to verify the instructions entered are the ones you think you entered
10. Use the trace command (t) to step through your program one step at a time and verify the program operation (type "t" then enter to execute each program step – inspect your register contents at each step to verify the program is operating correctly)
11. When you have reached the address in step 7 you program is complete
12. Use the *dump command* to verify your final result is at 2000:1000

Instructor Verification