NAME

# CIS40/CNET220 EXERCISE 2B
# 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 setting an offset into the SI register: (note your offset value is different)

```
MOV SI,0100
```

The above segment and offset form an address of 4000:0100, the below example shows how to Write the value 1000 stored in AX to memory address 4000:0100 using Register indirect mode

```
MOV [SI], AX
```

Reading a value from memory is just the opposite (that is [SI] and AX are reversed in order in the statement)

## PROBLEM STATEMENT

Summary: At the *default* CS and IP address assemble a program that will write F1D0 h to 2000:0100 then read it back, then write it to 3000:1000. Note for the default assembly address enter “a” only in debug

1. Go into the assembler (a) at the default address.
2. Turn the following steps into assembly language and enter the instructions:
3. Read F1D0 to the AX register using an *immediate* mode move statement
4. Write the AX register (with F1D0) to address 2000:0100 using the DI register in *RegisterIndirect* mode
5. Read address 2000:0100 to register BX using the DI register in *Register Indirect* mode
6. Write the contents of BX register to address 3000:1000 using the SI register in *Register Indirect* 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) 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 3000:1000

Instructor Verification