CS221 Assembly Language

Lab 3

Adapted from lab 2 by Prof. Mohamed Ben Othman

Practice:

Data Transfer:

Use the given program template to declare the variety of data transfer we studied in 04 part 1-IntroAssembly.

```
TITLE Flat Memory Program Template   (Template.asm)

; Program Description:
; Author:                    Creation Date:
; Modified by:               Modification Date:

.686
.MODEL FLAT, STDCALL
.STACK

INCLUDE Irvine32.inc
.DATA
    ; (insert variables here)
.CODE
main PROC
    ; (insert executable instructions here)
    exit
main ENDP
    ; (insert additional procedures here)
END main
```

Use your slides to:

A- Declare a `byte`, `word` and `dword` variables.

B- Use the `mov` instruction to move data from:

   a. Register to register (try all registers types: general-purpose and segment registers)
   b. From memory to register
   c. From register to memory
   d. Move an immediate value to register and to memory.

Note: you should know the restrictions (same operands' sizes, no memory to memory, no immediate values in destination).

C- Use the DUP operator to create arrays of `byte`s, `word`s, and `dword`s. Use the debugger to inspect the memory of these arrays. (See slide 26)

Exercises:

Inspect the memory and registers after each task.

D- Define symbolic constants using "`=`" and "`equ`" and `mov` them to different registers (See slides 33-34). Use the values:
   a. Hex value: FFFFFFF5
   b. Binary value: 10101010
   c. Octal value: 12345
   d. Hex value: ABCDEF0123456789

E- Can you redefine the names used in task D?

F- Use the `offset` operator to get the addresses of the arrays defined in task C. (See slide 36)

G- Use the `type` operator to find the size of an element in the arrays defined in C. (See slide 37)

H- Use the `lengthof` and the `sizeof` operators to find the number of elements and the size of the arrays defined in task C. (See slides 38-40)

I- Define variables of type `qword` and `tbyte`. (See slide 28)
   a. Can you `mov` their values to any of the registers? What about some of their values? Use the `ptr` operator. (See slide 41)
   b. Can you store in them the values defined in task D.