# Comp 3350: Computer Organization & Assembly Language

# HW # 6: Theme: Arithmetic and Procedures

*All main questions carry equal weight.*

*(Credit awarded to only those answers for which work has been shown.)*

1. [Arithmetic Expression] Write a program that computes the following arithmetic expression:

EDX = Dog + Cat - Mouse + Horse

Use the following data definitions:

Dog SWORD 8

Cat SWORD -25

Mouse SWORD -36

Horse SWORD -102

.386

.model flat,stdcall

.stack 4096

ExitProcess proto,dwExitCode:dword

.data

Dog SWORD 8

Cat SWORD -25

Mouse SWORD -36

Horse SWORD -102

.code

main proc

movzx edx, Dog

movzx eax, Cat

movzx ebx, Mouse

movzx ecx, Horse

add edx, eax

sub edx, ebx

add edx, ecx

invoke ExitProcess,0

main endp

end main

1. [Arrays] Write a program that:
2. Prompts the user for integer input 15 times
3. Stores these inputs in an array
4. Displays the stored array values on the screen using WriteInt (not DumpRegs).

In your submission, please embed the full program (.lst file) and one screen shot with at least one positive and one negative input value. Use the following:

.data

PromptUser BYTE "Please enter a value:", 0

Oranges SWORD 15 DUP(?)

A screenshot of a computer

Description generated with very high confidence

\*Note the that first value was cut off

1. [Compares, Procedures] Write a procedure, *MinMax* that computes the next to minimum and next to maximum values stored in the array *Oranges*. Write a main program that calls *MinMax* and prints the values found. Also, print the array indices at which the values were found.

Use the following:

.data

prompt BYTE "Please input a value: ", 0

spacing BYTE ", ",0;

Minimum BYTE "The minimum value of inputs is: and is located at index: ",0

Maximum BYTE "The maximum value of inputs is: and is located at index: ",0

In your submission, please embed the full program (.lst file) and one screen shot showing the values found.

Example:

Oranges SWORD 30, 60, 10, 40, 50, 20, 90, 100, 70, 80

Next to minimum 20

Next to maximum 90

Please test with positive and negative values

A screenshot of a computer

Description generated with very high confidence

1. Use a loop with indirect or indexed addressing to swap the odd indexed elements of an integer (word sizes) array in place as they occur. Please print the original and the swapped arrays. Please use several arrays with varying lengths to test your program.

Example:

Indices: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

MyArray SWORD 10, 20, 30, 40 ,50, 60 ,70, 80 ,90 100

After Swaps:

MyArray SWORD 10, 40, 30, 20, 50, 80, 70, 60, 90, 100

Please note that the last index 9 value does not need to be swapped.

A screenshot of a computer

Description generated with very high confidenceA screenshot of a computer

Description generated with very high confidenceA screenshot of a computer

Description generated with very high confidenceA screenshot of a computer

Description generated with very high confidence