WEEK: TOPIC: ~~Exercise~~ DATE:

Okpala Chukwuemeka
Mid-Term Examination
COE 306.

1a Because the addresses when ~~writing~~ coded in the instructions would have to be updated whenever new variables were inserted before existing ones.

1b Object.OBJ and Listing.LST files

2a Pro portability in high-level computer programming is the usability of the same software in different environments.

2b No. Each assembly language is based on either a processor family or a specific computer.

2c

32bit registers	31	16 15	8 7	0	16-bit registers
EAx		AH	AL	Ax	Accumulator
EBx		BH	BL	Bx	Base
ECx		CH	CL	Cx	Counter
EDx		DH	DL	Dx	Data

Ax is the primary accumulator
Bx is known as the base register.
Cx is known as the count register.
Dx is known as the data register.

WEEK: TOPIC: DATE:

3b Main proc: This identifies the beginning of the code.
MOV Ax, 47104 - This tells the program to move '47104' into the register Ax
Add EAx, 127Q - This tells the program to ~~move~~ the add 127(in base) to the value already existing in EAx register.
MOV Ds, Ax - This tells the program to move the value in Ax in to Ds
Main ENDp - This is the exit statement that calls a predefined ms-windows function that halts the program.

3C: Value 1 Byte 6Dh ; This tells the System to Store byte 6Dh under value 1 label (it is an unsigned byte).
label → Directive → Initalizer → radix

ii) value 2 DWORD ? - This is an uninitalized variable and it's value will be assigned at runtime.

iii) Value 3 SBYTE -10, -20, -30, -40, -50
These are multiple initalizers (Signed variables)
Here one label is used to declare multiple Signed variables

WEEK: 4 TOPIC: ~~extra~~ DATE:

```
TITLE Subtraction (Sub.asm)
; This program subtracts 3 16-bit integers
INCLUDE Irvine32.inc
.data
val1 WORD 5000h
val2 WORD 2000h
val3 WORD 1000h
final val ?
.code
main pROC
mov eax, val1
Sub eax, val2
Sub eax, val3
mov final val
DumpRegs
exit
main ENDP
END Main.
```