

EMMANUEL ELEGBE
COMPUTER ENGINEER
17/ENG102/020
COE 306

25th/07/2020

[Signature]

1a) It would not be a good idea to use numeric addresses when writing instructions that access variables because every time the addresses would be coded in the instruction would have to be updated when ever new variables were inserted before existing ones

b) The types of files produced by the assembler are called object file (.obj) and listing file (.lst)

2a) Portability is when a program can be moved to another operating system with (one operating system to another) with minor changes in code

2b) No it is not because the different chipsets are manufactured on different architectures

32-bit registers		16	15	8	7	16 bit registers	
2c) ↓ EAX			AH		AL		Ax accumulator
EBX			BH		BL		Bx Base
ECX			CH		CL		Cx counter
EDX			DH		DL		Dx Data

Ax: Primary accumulator

Bx: This is the base Register

Cx: This is the known count register

Dx: This is the data register

Git

3.8) Main proc: This directive indicates the beginning of a procedure

MOV AX, 4704: The MOV instruction copies the integer 47, 04 to the AX Register

ADD EAX, 1270: The ADD instruction adds 1270 to the EAX register

MOV DS, AX: This tells the program to move the value in AX into DS

Main ENDP - The main ENDP is the Bait statement indirectly calls a procedure MS-windows function that halts the program.

3a) When an assembly language program is assembled, the translation that the assembler carries out is essentially the construction of the text segment and data segment.

During the time the assembler is reading through and translating an assembly-language program, it always keeps track of where it is currently assembling to

C value1 Byte 6h - This tells the system to store byte 6h under value1 label
↓
Directive initialization

value2 now? - This is an uninitialized variable and its value will be assigned at runtime

TITLE Add AND subtract

~~Giff~~

: Program Assignment 1

code

main proc

INCLUDE Irvine32.inc

code

main proc

mov eax, 90000h

: EAX = 10000h

sub eax, 40000h

EAX = 50000h

sub eax, 20000h

EAX = 30000h

call DumpRegs

; display registers

Exit