

ASSIGNMENT # 01

Date: 09/10/21

Question # 01:

Explain the roles of compiler and OS in executing a HLL (High level language) program from a text file. Why is the process of linking performed.

Answer:-

When a High level language is compiled, the compiler converts the source code into Assembly language (Machine Readable form). The Assembler then converts it into binary code (0's and 1's). The OS loader takes this executable file created by assembler to the main memory and branches the CPU towards its starting address and the program is then executed. The linker is used to link procedures from a library that is called within the program.

Question # 02:

Explain the contents of Segment registers in Real memory addressing mode and Protected mode.

The segment registers hold pointers to memory segment. In real memory addressing mode, it points to base addresses of pre assigned memory areas.

Date: _____

In protected mode, the segment registers hold pointers to descriptor table.

The descriptor table describes the location, length and access rights of memory segments.

Question #03

a) Segment: $ABDEH$

Offset: $5D89H$

Real address: ?

$$\begin{array}{rcl} \text{Segment} \times 10 & = & ABDE0 \\ + \text{offset} & + & 5D89 \\ \hline & & BDE69 \end{array}$$

b) Segment: $8FE3H$

Offset: ?

Real address: $A835FH$

$$\begin{aligned} \text{Offset} &= (\text{Real address}) - (\text{Segment} \times 10) \\ &= A835F - 8FE30 \end{aligned}$$

$$\begin{array}{r} \begin{array}{cccccc} & 9 & & 7 & & \\ A & 8 & 3 & 5 & F & \\ - & 8 & F & E & 3 & 0 \\ \hline 1 & 8 & 5 & 2 & F & \end{array} \end{array}$$

1) Segment: 7

Offset: 5E6DH

Real Address: FF41DH

Solⁿ:

$$R. Address = (Segment \times 10) + Offset$$

$$\frac{R. Address - Offset}{10} = Segment$$

$$\begin{array}{r}
 FF41D \\
 - 5E6D \\
 \hline
 F95B0 \\
 \div 10 \\
 \hline
 F95B
 \end{array}$$

The Segment Address is F95B.

2) Segment: ?

Offset: ?

Real Address: A5B6DH.

Since we have the real address therefore we can suppose values of Segment & Offset.

Let Segment

Offset = 1A7C

Offset = 2FD

Question #04

Explain in your own words, why .class file of JAVA is platform independent; whereas executable files (Machine code) generated on Java is platform independent

When a program written in Java is compiled. The compiler converts it into byte code (.class) file. This byte code is platform independent because ~~the JVM~~ because it is same for the similar OS.

This byte code is converted into Machine code by JVM (Java Virtual Machine). The JVM is platform dependent and it converts the given byte code into the machine readable form and has different implementation for different OS.

Question # 05

Discuss the similarities and difference between real address mode and virtual-8086 mode.

Similarities:-

The real address mode & 8086 mode gives only 1 MIB of accessible memory to the programs for execution.
For E.g. Programs like MS-DOS.

Difference:-

In real address mode, the programs can access any part of memory while virtual 8086 mode is a special type of protected mode where only 1 MIB of memory is accessible for the program.

In real address mode, the program can cause the Operating system to stop responding to commands whereas in 8086 virtual mode the program crashes but Operating system does not.

Question #16

Explain the purpose of control flags and status flags.

Control flags:-

Control flags determine how instructions are carried out. These flags include:

Direction flag:- affects the direction of block data transfer

Interrupt flag:- Determine whether interrupt can occur.

Trap flag:- Determine whether CPU is halted after every instruction.

Status flags:-

These flags reflect the outcomes of different arithmetic operations. These include

- i) Carry flag: shows if there exists a carry
- ii) Overflow flag: when the result is large (beyond)
- iii) Sign flag: show if the number is negative or positive
- iv) Zero flag: shows if the result is zero
- v) Auxiliary flag: shows if there is a carry below
- vi) Parity flag: show if the result is even parity or not.