|  |  |
| --- | --- |
| **SESSION** | **FEB-MARCH 2025** |
| **PROGRAM** | **BACHELOR OF COMPUTER APPLICATIONS (BCA)**  |
| **SEMESTER** | **IV** |
| **COURSE CODE & NAME** | **DCA2203 SYSTEM SOFTWARE** |
|  |  |
|  |  |

**Set-I**

**Q1. Describe the function and significance of the segment and pointer registers in the 8086 microprocessor. How do these registers support memory segmentation and data handling?**

**Ans 1.**

**Segment and Pointer Registers in 8086 Microprocessor**

**Function and Significance of Segment Registers**

The 8086 microprocessor operates on a segmented memory model, allowing it to access 1MB of memory using 20-bit addressing, despite having only 16-bit registers. This is achieved through the use of segment registers, which divide memory into four segments: Code Segment (CS), Data Segment (DS), Stack Segment (SS), and Extra Segment (ES).

* **Code Segment (CS)** holds the code or instructions the processor executes. It points to the
* **Stack Segment (SS)** handles the stack, which stores return addresses, parameters, and

Its Half solved only

Buy Complete assignment from us

**Price – 190/ assignment**

**MUJ Manipal University Complete SolvedAssignments MARCH 2025**

buy cheap assignment help online from us easily

we are here to help you with the best and cheap help

**Contact No – 8791514139 (WhatsApp)**

**OR**

**Mail us-** bestassignment247@gmail.com

**Our website -** [www.assignmentsupport.in](http://www.assignmentsupport.in)

**Q2. Differentiate between the roles of the Symbol Table and the Literal Table in an assembler. Describe how each is constructed during Pass 1 and how they are used during Pass 2.**

**Ans 2.**

**Symbol Table vs. Literal Table in an Assembler**

**Role of the Symbol Table in an Assembler**

In the process of translating assembly code into machine language, the symbol table plays a critical role in tracking identifiers (labels or variables) and their corresponding memory addresses. A symbol in assembly refers to a name assigned to a memory location, instruction, or data item. These symbols are stored in the symbol table along with relevant attributes such as

**Q3. Compare and contrast Absolute Loaders and Relocating Loaders. Why are relocating loaders preferred for advanced systems? Support your answer with examples and scenarios. 5+5**

**Ans 3.**

**Absolute Loaders vs. Relocating Loaders**

**Absolute Loaders**

An absolute loader is the simplest type of loader used to load executable code into memory at a specific fixed address. The loader reads the object code generated by the assembler and directly places it at the assigned memory location. The object program contains memory addresses that are fixed during the assembly process. No address modification is done by the loader at runtime.

For example, if the assembler sets a program to load at memory address 2000H, the absolute

**Set-II**

**4. Compare and contrast device driver management in UNIX/Linux, MS-DOS, and Windows operating systems. Discuss the key differences in installation, configuration, communication, and architecture. 5+ 5**

**Ans 4.**

**Comparison of Device Driver Management in UNIX/Linux, MS-DOS, and Windows**

**Device Driver Architecture and Communication**

Device drivers are essential software components that allow the operating system to interact with hardware devices. The architecture and handling of device drivers vary significantly across UNIX/Linux, MS-DOS, and Windows operating systems.

**Q5. Explain the process of IP Address allocation in UPnP devices using both DHCP and Auto-IP mechanisms. Discuss the importance of address management in UPnP and how it ensures seamless device communication on the network. 5+5**

**Ans 5.**

**IP Address Allocation in UPnP Devices**

Universal Plug and Play (UPnP) is a protocol suite that enables devices to automatically discover each other and establish functional network services. For this to happen smoothly, devices must be assigned unique IP addresses within the network. UPnP relies on two main mechanisms for IP

**Q6. Explain the hierarchy process in Android memory management. How does Android decide which process to terminate when memory is low? 5+5**

**Ans 6.**

**Hierarchy and Process Termination in Android Memory Management**

**Android Memory Management**

Android is designed to run on mobile devices with limited memory resources. To ensure smooth operation, it implements a hierarchical memory management system that classifies processes based on their importance to the user and the system. This hierarchy guides how Android allocates and reclaims memory, especially under low-memory conditions.

**Process Hierarchy in Android**

Android divides processes into several categories, ranked from most to least important: