Biyani Institute of Science & management

I Internal Examination Sep. 2019

Class: MCA-V

Subject- Data Intensive computing and Security (MCA-505A)

MM: 20 **Set: B** **Time: 1½Hrs**

[I] Answer the following questions in one line only **(2*1=02)**

Q.1 What is Process addressing?

Ans) Addressing modes are an aspect of the instruction set architecture in most central processing unit (CPU) designs. The various addressing modes that are defined in a given instruction set architecture define how machine language instructions in that architecture identify the operand(s) of each instruction. An addressing mode specifies how to calculate the effective memory address of an operand by using information held in registers and/or constants contained within a machine instruction or elsewhere.

Q.2 What is Thread based Parallelism?

Ans) A thread in computer science is a set of instructions that can be managed independently by a scheduler, which is a part of operating system.The main function of Threading is to run multiple threads at a time. Threads means different tasks, function calls in the program and multiple threads run at the same time that does not means that they are executed on different machines.

[II]Answer the following questions in 50 words **(2*3=06)**

Q.1 Explain the Parallel programming Design?

Ans) In computing, a parallel programming model is an abstraction of parallel computer architecture, with which it is convenient to express algorithms and their composition in programs. The value of a programming model can be judged on its *generality*: how well a range of different problems can be expressed for a variety of different architectures, and its *performance*: how efficiently the compiled programs can execute. The implementation of a parallel programming model can take the form of a library invoked from a sequential language, as an extension to an existing language, or as an entirely new language.Consensus around a particular programming model is important because it leads to different parallel computers being built with support for the model, thereby facilitating portability of software. In this sense, programming models are referred to as *bridging* between hardware and software.[2]

Q.2 What is Synchronization?

Ans) Synchronization, in the context of .NET, is a process that involves coordinating the execution of multiple threads to ensure a desired outcome without corrupting the shared data and preventing any occurrence of deadlocks and race conditions.Synchronization also occurs between network nodes to ensure that data streams are received and transmitted correctly, and to prevent data collision. It usually uses a clock signal transmitted in sequence with a data stream to maintain proper signal timing.

[III]Answer the following questions in 150 words. **(2*6=12)**

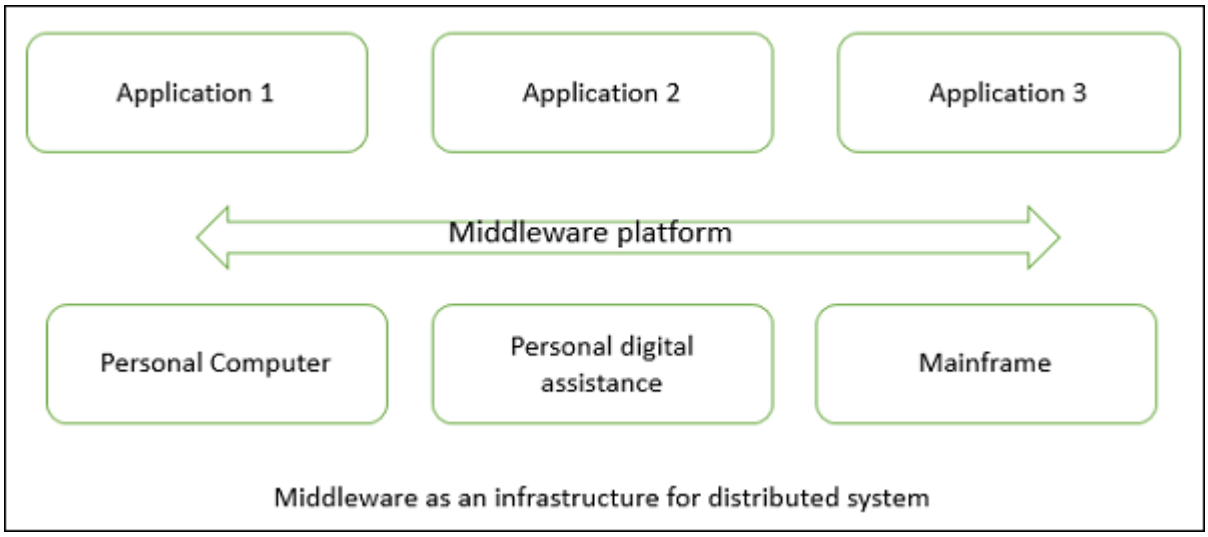
Q.1 Explain the Distributed computing Architecture?

Ans) Distributed computing is a computing concept that, in its most general sense, refers to multiple computer systems working on a single problem. In distributed computing, a single problem is divided into many parts, and each part is solved by different computers. As long as the computers are networked, they can communicate with each other to solve the problem. If done properly, the computers perform like a single entity.

The ultimate goal of distributed computing is to maximize performance by connecting users and IT resources in a cost-effective, transparent and reliable manner. It also ensures fault tolerance and enables resource accessibility in the event that one of the components fails.

- In this architecture, information processing is not confined to a single machine rather it is distributed over several independent computers.
- A distributed system can be demonstrated by the client-server architecture which forms the base for multi-tier architectures; alternatives are the broker architecture such as CORBA, and the Service-Oriented Architecture (SOA).
- There are several technology frameworks to support distributed architectures, including .NET, J2EE, CORBA, .NET Web services, AXIS Java Web services, and Globus Grid services.
- Middleware is an infrastructure that appropriately supports the development and execution of distributed applications. It provides a buffer between the applications and the network.
- It sits in the middle of system and manages or supports the different components of a distributed system. Examples are transaction processing monitors, data convertors and communication controllers etc.

Middleware as an infrastructure for distributed system



Middleware as an infrastructure for distributed system

Q.2 Explain the Following terms:

(a) Parallel Algorithm

Ans) A parallel algorithm can be executed simultaneously on many different processing devices and then combined together to get the correct result. Parallel algorithms are highly useful in processing huge volumes of data in quick time. This tutorial provides an introduction to the design and analysis of parallel algorithms. In addition, it explains the models followed in parallel algorithms, their structures, and implementation.

(b) Parallel Metrics

Ans) • For the purpose of computing speedup, we always consider the best sequential program as the baseline

For the purpose of determining how effective our parallelization technique is, we can determine a pseudo-speedup w.r.t. the sequential version of the parallel algorithm

• Speedup can be as low as 0 (the parallel program never terminates).

• Speedup, in theory, should be upper bounded by p - after all, we can only expect a p-fold speedup if we use times as many resources.

• A speedup greater than p is possible only if each processing element spends less than time TS / p solving the problem.

• In this case, a single processor could be timesliced to achieve a faster serial program, which contradicts our assumption of fastest serial program as basis for speedup.