Name: Sandesh Thapa Magar
Branch: Master of engineering in Computer Engineering
Rollno:242959
Subject:Advanced Computer Architecture
Assignment-1:UNit-1(5_question_answer)

1) Explain & Compare multitasking, multiprocessing & multithreading.

Ans

Multiprocessing	Multithreading	Multitasking
1) The availability of more than one processor per system which can execute several set of instructions in parallel is called multiprocessing	A process is divided into several different subprocesses called threads which has its own path of execution is called multithreading	The execution of more than one task simultaneously is called multitasking
2) Less time is taken for job processing time	2) Moderate amount of time is taken for job processing	2) Moderate amount of time
3) Number of CPU is more than one	3) It can be one or more than one CPU number	3) Number of CPU is one
4) Threughput is ~~less~~ maximum	4) Throughput is ~~maximum~~ moderate	4) Throughput is moderate
5) It's efficiency is maximum	5) It's efficiency is moderate	5) It's efficiency is moderate
6) It categorizes into Symmetric & Asymmetric	6) It No further division	6) It categories into ~~one~~ single users & multi user

Page
Date

2) Define parallel processing and discuss its significance in modern computing environments

Ans

→ Parallel processing refers to the process of executing several processors an application or computation simultaneously. Generally.

→ It is kind of computing architecture where large problems break into independent, smaller, usually similar parts that can be processed in one go.

→ It is done by multiple CPUs communicating via shared memory which combines result upon completion

It's significance in modern computing environment

→ It is key to make data more modeling, dynamic simulation and for achieving the same. Therefore, it is needed for the real world too

→ Only the concept of parallel processing can organize large datasets, complex & their management

→ It provides surety the use of resources effectively and guarantees the effective use of hardware whereas only some parts of hardware are used in serial computation & some parts are rendered idle.

→ It offers concurrency & saves time & money.

3) What are primary challenges associated with implementing parallel processing system? Discuss trade-offs between complexity, cost and performance in parallel computer architecture design

Ans

The primary challenges associated with implementing parallel processing are:

1) Synchronization overhead
2) Data Dependency & communication overhead
3) Load balancing
4) Memory Bottlenecks
5) Scalability Issues
6) Fault tolerance & Reliability
7) Programming challenges

Trade-offs between complexity, cost & performance

1) Power consumption:
   → High-performance parallel systems consume more power, increasing operational costs and necessitating better cooling solutions

2) Memory Hierarchy:
   → adding multiple levels of cache & ensuring coherence improves performance but increases complexity & cost.

3) Cost vs performance:
   → Higher performance interconnects & large shared memories boost performance but are expensive

Page
Date

4) Complexity vs Cost:
→ Simplifying architectures lower costs but may degrade performance
⇒ Highly optimized architectures are costly to design & manufacture but deliver significant performance

5) Complexity vs performance
→ Increasing parallelism can enhance performance but also increases system complexity.

4) Compare & contrast Flynn's taxonomy with feng's classification of parallel computers. How does feng's framework enhance the understanding of parallel processing system

Ans
⇒ Flynn's Taxonomy and feng's classification are two approaches to categorizing parallel computers but they differ significantly in focus and application.

1) Flynn's Taxonomy:
It is classified computers based on instruction streams and data streams. It includes four main categories:
i) SISD (Single Instruction, Single Data):
→ Traditional sequential computers
→ Eg: standard uniprocessor system
ii) SIMD (Single Instruction, Multiple Data)
⇒ Executes same instruction on multiple data simultaneously.
⇒ Eg: Vector processor, GPU

Page
Date

MISD (Multiple Instruction, Single Data):
→ Multiple instruction operate on the same data
→ Rarely implemented in practice
→ Eg: Some fault-tolerant systems

MIMD (Multiple Instruction, Multiple Data):
→ Independent processors execute different instruction on different data sets
eg: Multi-core CPU, distributed systems

Feng's classification
It introduces a granularity-based approach by analyzing parallel systems based on their degree of parallelism and granularity of operations. It considers the size of operation and communication costs btw processors, leading to three main types:

Word-level parallelism:
focuses on parallelism within single word or register. eg: processors with SIMD instruction.

Bit-level parallelism:
→ Exploits parallelism within individual bits of word. System with wide data path for specific operations

Task-level Parallelism:
→ operates on distinct tasks or processes concurrently
Eg: System using MIMD architecture

Page
Date

Comparision

Feature	Flynn's Taxonomy	feng's classification
focus	Instruction & data streams	Granularity & degree of parallelism
Granularity	Not explicitly addressed	core concept (bit, word, task level)
Scalability	Broad categories with limited detail	includes measures like DoP, CR
Practical Use	High-level categorization	More detailed for performance analysis
Applicability	General classification of architectures	Evaluation of efficiency & design trade-off

feng's frameworks enhance ~~wabt~~ ~~unst~~ understanding of parallel processing by

1) focus on Efficiency
2) Granularity levels
3) Scalability & performance analysis
4) Design insights
5) Hardware & Software Synergy.

Compare MPP & cluster Computing

MPP (massively Parallel Processing)	Cluster Computing
It is tightly-coupled system with hundreds or thousands of processors working together on single problem	1) It is lossely coupled system consisting of group of independent computers connected via network
lower latency due to specialized interconnects	2) Higher latency due to standard networking
Requires specialized knowledge for maintenance & upgrades	3) Easier to maintain due to widely available components & software
optimized for specific tasks	4) Flexible & suited for diverse network
limited fault tolerance	5) Higher fault tolerance.