PP Theory Assignment-1

Priyal Agrawal
H2 Batch
Roll No -27
Exp-10322014O6

1) Explain superscalar Architecture and elaborate the type of dependencies that affects superscalar Architecture.

→ Superscalar architecture is a method of parallel computing used in many processors. In a superscalar computer, the CPU manages multiple instructions pipelines to execute several instructions concurrently during a clock cycle. Superscalar design involves the processor being able-to use multiple instructions in a single clock, with redundant facilities to execute an instructions.

The types of dependencies that affect super-scalar architecture are:-

(i) True data dependancies - The result of one operation is an input to next.

(ii) Resource dependancy - Two operations require the same resource.

(iii) Branch dependancy - Scheduling instruction, across conditional branch statements cannot be done deterministically a-priori.

The schedular, a piece of hardware looks at a large number of instructions in an instruction queue & selects appropriate number of instructions to

execute concurrently based on these factors. The complexity of this hardware is an important constraint on superscalar processors.

2) Discuss the application of High performance computing in atleast 3 different domains.

Ans (i) Health care - Medicines and computing are interwined. Computers store confidential patient information, track vital signs and analyze drug efficiency. The rise of HPC has allowed medical professionals to digitize even more complex processes like geneme sequencing and drug-testing.
Eg → Living heart project - A cyber Heart, etc.

(ii) Engineering - Engineering is all about boosting a machine's real-world performance, but testing prototype is expensive. To work around this, engineers often test meme designs in massive computers simulations unlike the real morlal, they run on HCP systems. So far, simulations have been used to test the functionality of airplane parts, stream line racing bike frames and much more.
Eg → Boeing: A lighter 787, etc.

(iii) Space Research - Outter Space is full of unknowns. The concept of aliens, collisions, whether of sun & other such subjects take act of resources & technological savy to gather data that's necessary to find answers. That's where model rooted in HCP

comes in handy. They make most of the information gleaned by probes & satellites.
Eg → Polarbear : The origin of everything, etc.

(iv) Finance and Business - HCP systems are essential and massively powerful computers. In cyptocurrency, context, HCP system essentially prints money.
Eg → BITcoin :- mining for an algorithmic boss, etc.

3) Elaborate the concept of granularity with example?

Ans Granuality of a task is a measure of the amount of work (or computation) which is performed by that task. The number of tasks into which a problem is decomposed determines its granuality. Decomposition into large number of tasks a results in fined grained decomposition and that into smaller no. of tasks results in coarse grained decomposition.
A coarse grained counterpart to the dense matrix-vector product example. Each task in this example corresponds to the computation of 3 elements of the result vector.

Example :-

```
for (i=0; i<n; i++)          // original
    a(i) = d(i) + b(i)/2
for (i=0; i<n; i + vectorsize) // vectorized
VR1 = Vector load (ai)
VR2 = Vector load (b, i)
VR! = VR1 + VR2/2
```

Vectorstore (a, vectorsize, VR1)

VR1	a(0)	a(1)	a(2)	a(3)
VR2	b(0)	b(1)	b(2)	b(3)

4) Explore if bubble sort can be parallely implement-ed? If not write report in detail & modification if needed?

→ Yes bubble sort can be implemented parallely. The concept of parallelism involves executing a set of instruction/code simultaneously. The main benefit of this is faster computation in parallel bubble sort, we divide sorting of the unsorted into 2 phases - odd & even. We compare all the pairs of elements in list/array side by side. When it is the odd phase, we compare the element at index 0 with the element at index 1, the element at index 2 with element at index 3, & so on. In even phase, we compare index 1 element with index 2 element & so on. While comparing, we shuffle the elements, if initial element is greater than the next element.

6	5	3	4	step 1 (odd)

5	6	3	4	step 2 (even)

Q. No.	1	2	3	4	5	6	7	8	9	10	Total
Marks/Grade											

Name & sign of the faculty Member

(Please start writing assignment/ test from here)

5	3	6	4

step 3 (odd)

3	5	4	6

step 4 (even)

3	4	5	6

In multicore, both phase occur simultaneously.