- GPU allocate a higher amount of cache as compared to CPU
- Branch handling in GPU is superier as compared to CPU
- GPU have larger chip area for functional/compute unit as compared to CPU

4

Which is statement is true for dynamic networks
(1 Point)

- Nodes are connected to each other directly
- (●) Dynamic networks with 2LogN stages are always non-blocking network
- Every node have their own switches and is connected to the network via a BUS
- Dynamic networks are costly and not used in the regular design of HPC system

5

Given a system of N nodes with a 2D Torus interconnection network. The system has a diameter, the total number of links, and bisection bandwidth
(2 Points)

- 2*Sqrt N, 4N, Sqrt N
- (●) Sqrt N, 2N, 2*Sqrt N
- Sqrt N, 4N, 2*Sqrt N
- log N, N, 2*Sqrt N

6

Explain the problem R2| |Cmax
(1 Point)

- Scheduling n independent tasks with arbitrary execution time on two identical processors t minimize the overall execution time.

- ○ Scheduling n independent tasks with arbitrary execution time on two unrelated processors to maxmize the overall execution time.
- ◉ Scheduling n independent tasks with arbitrary execution time on two unrelated processors to minimize the overall execution time.
- ○ Scheduling n independent tasks with arbitrary execution time on two unrelated processors to minimize the overall waiting time.

7

CP/HLF scheduling is optimal P|pj=1,out-tree|Cmax with time complexity O(V+E), where V and E are number of node and edges of the tree. Suppose we want to use the same CP/HLF to solve P| pj, out-tree|Cmax, by simply converting each node with pj execution time to a series of pj nodes with unit execution time and then apply the CP algorithm. This will produce an optimal result but the time complexity of the CP on the transformed out-tree will be (Assume W= Σpj)
(2 Points)

- ○ O(V+E), it is polynomial time algorithm
- ○ O(V+E)+W, it is a Pseudo polynomial time algorithm
- ○ O(V+W+E+W), it is a Pseudo polynomial time algorithm
- ○ O(V+W^2+E), it is a Pseudo polynomial time algorithm

8

Can be the problem 1|pj=1| ΣTj be solved
(1 Point)

- ○ Optimally in Polynomial time using earlier deadline first (EDF) rule
- ○ Problem is very difficult to solve polynomially but EDF is good heuristics for the same
- ○ We cannot say
- ○ All of the aboves are correct statement

9

Characteristics of collective communications in MPI Program, tick the correct option
(1 Point)

- (●) Collective communication can always be act as barrier synchronization for the processes
- ( ) All the processes participate in the collective communication
- ( ) Some special processes are excluded from the collective communications
- ( ) All_scatter and All_gather are collective communication but scatter and gather are not collective communication.

10

Why vector sum application is not a good candidates for GPU acceleration?

```
for(i=0;i<N;i++) A[i]=B[i]+C[i];
```

(1 Point)

- ( ) Vector sum require less computation per data
- ( ) Vector sum is data intensive application and it is not cache friendly
- (●) Vector sum require Locking and Unlocking protocol in GPU cores
- ( ) All the above

11

Suppose there are 5 identical processors and LPT rule is used to schedule the independent tasks with arbitrary execution without pre-emption to minimize Cmax, the achievable approximation can be _____. (Ans in one word/numeric value)
(2 Points)

19/15

12

Programming model for GPU: tick the wrong one
(1 Point)

- In GPU programming, address space seen by the host processor and the GPU are different
- Inside the GPU, all the tiny cores see the same address spaces
- Different SM of the same GPU, see the different address space
- Thread blocks gets scheduled to SMs.

13

CP/HLF Scheduling Approximation for Pm|pj=1, prec|Cmax for two processors proven to be achieved __approximation. (Ans in one word/numeric value)

(2 Points)

4/3

14

Choose the right explanation of the problem Q|pmtn, di, ri, pj|ΣUj
(2 Points)

- Minimizing the number of missed tasks for tasks with infinite deadlines, release time, arbitrary execution time, pre-emption allowed on Identical processors
- (●) Minimizing the number of missed tasks for tasks with deadlines, release time, arbitrary execution time, pre-emption allowed on Uniform processors
- Minimizing the number of missed tasks for tasks with deadlines, online tasks, arbitrary execution time, pre-emption allowed on Uniform processors

- ○ Minimizing number of missed tasks for tasks with deadlines, release time, arbitrary execution time, pre-emption allowed on Identical processors

15

What is the status of the problem Q|pj|ΣCj
(2 Points)

- ○ Can not be solved optimally using polynomial algorithm
- ○ Can not be solved polynomially and but 2-Approximation is available with List scheduling
- ◉ Can be solved optimally in polynomial time.
- ○ We cannot say.

16

Given an parallel application with serial fraction value = 0.25 and parallel fraction is divisible load. Calculate the maximum achievable speed up even if we are using infinite number of processor.
(2 Points)

4

17

Which is not part of MPI system management
(1 Point)

- ○ Process management
- ○ Remote memory transfer over the network
- ○ Virtual memory management
- ◉ All of the above

18

Suppose an application performs (a) broadcast, (b) scatter, and (c) gather collective operations from one process most of the time during the execution of the application. Given the option of interconnection network with the minimum number of links for the application, which interconnection network is preferable from the performance point of view.
(2 Points)

Star

19

Among these networks, which network has the highest bisection bandwidth
(1 Point)

- ○ Fat Tree
- ○ 2D Torus
- ◉ Hypercube
- ○ Star

20

Which is a valid assumption behind Amdhal's Law
(1 Point)

- ○ Parallel section can have lock-unlock structure
- ○ Processors can have their own cache memory
- ○ All the processors can be of any type from P, Q, and R
- ◉ All the parallel sections need to be purely parallel

21

Calculate the load factor, dilation, and congestion for the embedding of 16 nodes mesh onto 4 node torus.
(2 Points)

- (●) 4, 1, 2
- ( ) 4, 2, 1
- ( ) 4, 1, 1
- ( ) 4, 2, 2

22

Characteristics of Applications that are suitable for GPU acceleration are
(1 Point)

- ( ) Application with dominated by data transfer or streaming access
- ( ) Application which is array intensive and have a lot of branches
- ( ) Parallel application requiring lock and unlock operations
- (●) Application with small independent function/code executed huge number of times and the application need to be compute intensive

23

Suppose we want to embed N-1 (assume N is some integer power 2) nodes tree network to an N node hypercube, what will be the value of dilation, load factor, and congestion for the optimal embedding
(2 Points)

- ( ) LogN, 1, 1
- ( ) LogN, 1, log N
- ( ) 1, 1, 1

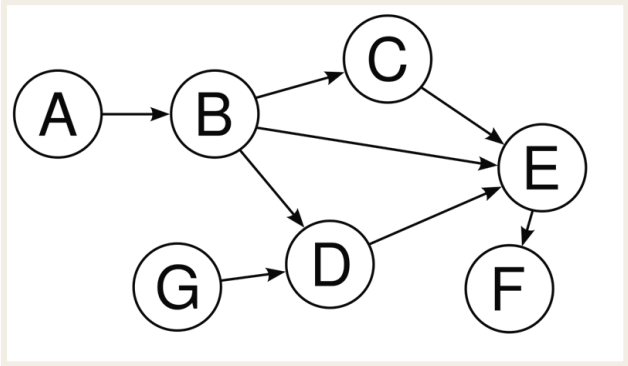
- [x] 2, 1, 1

24

Can we solve Pm|pj=1|ΣwjUj in polynomial time?
(2 Points)

- [ ] Can not be solved polynomially.
- [ ] Can not be solved polynomially and 2-Approximation is available with List scheduling
- [ ] Can be solved optimally in polynomial (with m and n) time.
- [ ] We cannot say.

25



Calculate span for the given DAG assume execution time of all the nodes are unit time.
(2 Points)

5

26

Choose the option which may not be the reason behind super-linear speed up
(1 Point)

- [ ] When running on P cores, the overall amount of cache may be higher

- ○ Failure to use the best uniprocessor algorithm as compared to parallel one
- ◉ Single processor may be a very old generation processor
- ○ All of the above

27

Given a system of N nodes, our aim is to interconnect them and our goal is to minimize the diameter by utilizing a minimum number of links to be used. Which interconnection network is preferable
(2 Points)

- ○ Fat-Tree
- ○ Tree
- ◉ Star
- ○ HyperCube

Submit