Quiz-1 Aug 31

Due 1 Sep at 23:55 **Points** 20 **Questions** 9
Available 31 Aug at 8:00 - 3 Sep at 11:00 3 days **Time limit** 90 Minutes

Instructions

This quiz covers basics of the concepts of parallel computing, properties of interconnect networks, and parallel algorithm design.

This quiz was locked 3 Sep at 11:00.

Attempt history

	Attempt	Time	Score
LATEST	Attempt 1	88 minutes	12 out of 20

Score for this quiz: **12** out of 20
Submitted 1 Sep at 23:32
This attempt took 88 minutes.

Question 1 — 2 / 2 pts

Suppose you are given a program which does a fixed amount of work and some fraction s of that work must be done sequentially. The remaining fraction of the work is perfectly parallelizable on P number of processors. Assuming T_1 is the time taken on one processor, derive a formula for T_p, the time taken on P processors.

Your answer:

s= serial fraction gamma

1-s= remaining fraction of work parallelizable

T1= time taken on one processor

P= number of processors

Therefore:

T_p=(s) T_1+(((1-s)T_1) / P)

$$T_p = T_1 s + \frac{T_1(1-s)}{P}$$

Question 2

1 / 1 pts

A program executes in 242 seconds on 16 processors. Through benchmarking it is found that 9 seconds is spent performing initializations and clean-up on one processor, i.e. it is the serial part of the computation. During the remaining 233 seconds all 16 processors are active. What is the scaled speedup achieved by this program?

Your answer:

Gamma scaled= 9/233

Scaled speedup= P+ (1-P) gamma scaled

=16(1-16)(9/233)

=15.42

The scaled speedup is $S \approx 15.4$.

Question 3

1 / 1 pts

A parallel program is executed on 40 processors. Benchmarking shows that the program spent 99% of its time inside parallel code. What is the

scaled speedup of this program?

Your answer:

1-Gamma=0.99

Gamma= 0.01

Gamma scaled= 0.01/0.99

Scaled speedup= P(1-P)Gamma scaled

=40(1-40)(0.01)

=39.61

$S = 40 + 0.01 \times (1 - 40) = 39.61.$

Question 4

0 / 1 pts

A computer is equipped with 4-core CPU. Each core runs at 1.3GHz clock rate, and can process up to 4 floating point operations within one clock cycle. What is the theoretical peak performance of this CPU in terms of FLOPS (floating point operations per second)?

Your answer:

5.2GFLOPS

It is $1.3 \times 10^9 \times 4 \times 4 = 20.8 \times 10^9 = 20.8$GFLOPS.

Question 5

1 / 2 pts

A serial program is parallelized to run using 4 processors. Benchmarking reveals that the sequential execution time is 100ms, while the parallel execution time is 55ms; that is, we have obtained a speedup of only $\frac{100}{55}$ or 1.8 instead of 4-fold speedup for which we may have hoped. Which of the following **factors** may contribute to this less than optimal speedup?

orrect answer

- [ ] C. The time the processor is stalled waiting for data to be communicated with another processing node

- [ ] D. The time the processor stalled waiting for a data to be loaded from the local memory

- [ ] A. The time that the processor spends executing instructions that are needed both in sequential and parallel programs

Correct!

- [x] B. The time that the processor spends executing instructions that are not needed in sequential program, but only in the parallel program

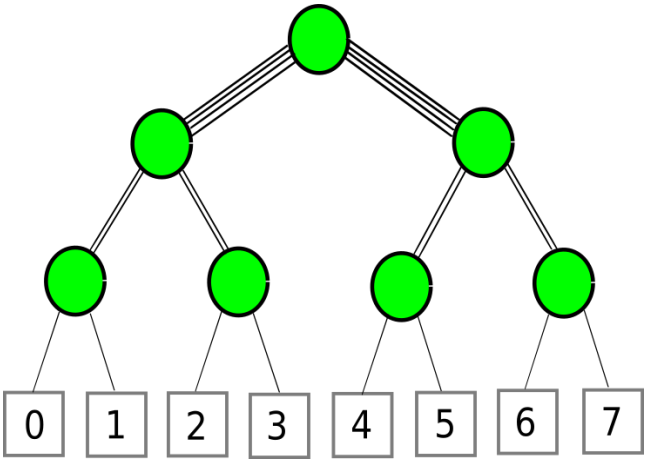
Correct!

- [x] E. The time that the processor spends waiting for another processor to complete certain data processing

Question 6

4 / 6 pts

What are the diameters of a 4-node ring, 4-by-4 2D mesh, and fat tree shown in the following figure, respectively? What is the bisection width for each of the above interconnects?



Your answer:

4 Node ring:

Diameter- 2

Bisection width-2

4 by 4 2D mesh:

Diameter- 6

Bisection width- 4

Fat tree:

Diameter- 1.69

Bisection width- 3.5

Diameter: ring, 3; 2D mesh, 6; fat tree, 6. Bisection width: ring, 2; 2D mesh, 4; fat tree: 1 switch or 4 links.

Question 7

1 / 3 pts

Consider a memory system with a DRAM of 512MB and L1 cache of 32KB with the CPU operating at 1GHz. The $l_{DRAM} = 100$ns and

$l_{L1} = 1$ns (l represents the latency). In each memory cycle, the processor fetches 4 words. Answer the following questions (show your reasoning or steps).

1. What is the peak achievable performance of a dot product of two vectors?
2. What is the peak achievable performance of a dot-product based matrix-vector multiplication assuming the size of the matrix is $4K \times 4K$ (each row of the matrix takes 16KB storage)?

Your answer:

1. Peak achievable performance= 4 x 1GHz/100ns which is equivalent to 4000000000FLOPS/ 100ns which can be simplified to 40MFLOPS

2. (80%)1+ (10%)100+10%(400)

=0.8+10+40

=50.8ns

1. The computation performs 8 FLOPS on 2 cache lines, i.e., 8 FLOPS in 200ns. This corresponds to a computation rate of 40 MFLOPS. the two cache lines are for the two input vectors respectively. Then the computation entails one multiply followed by one addition for each pair of input elements, thus, 8 FLOPS for each 2 cache line loads. loading each cache line takes 200ns, thus 8FLOPS in each 200ns.
2. In the best case, the vector gets cached and will be reused. One cache line is still 4 words, hence 8FLOPS can be performed on each cache line (for fetching the row vectors of the matrix). This corresponds to a peak performance rate of 80MFLOPS ($8\text{FLOPS}/100\text{ns} = 8 \times 10^7 = 80MFLOPS$)

Question 8

1 / 1 pts

Many parallel algorithms for distributed memory systems require a broadcast step in which one task communicates a value it holds to all of the other tasks. Which of the following statements is true?

D. An interconnect network with a smaller bisection width means a faster broadcast.

Correct!

A. An interconnect network with a smaller diameter means a faster broadcast.

C. An interconnect network with a smaller cost means a faster broadcast.

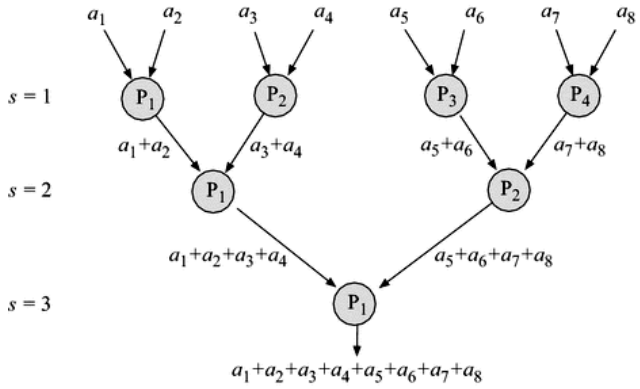
B. An interconnect network with a smaller diameter means a slower broadcast.

Question 9

1 / 3 pts

The attached figure 'tree_sum.png' shows an example of adding 8 numbers in parallel using 4 processes. Answer the followig questions.

1. What is the minimum number of processing elements you need to use to add n numbers using this approach?
2. What is the potential speedup you can achieve using this parallelization approach to add n numbers? Explain.



Your answer:

1. ceil((n/2))

2. n/P

1. $n/2$.
2. The sequential sum takes $O(n)$. The parallel sum takes $\lceil \log n \rceil$ or $O(\log n)$ time. Hence the potential speedup is $O(\frac{n}{\log n})$ or $\frac{n}{\log n}$.

Quiz score: **12** out of 20