Question Paper

Exam Date & Time: 27-May-2023 (02:30 PM - 05:30 PM)

MANIPAL ACADEMY OF HIGHER EDUCATION

VI SEMESTER B.TECH END SEMESTER EXAMINATIONS, MAY 2023

PARALLEL PROGRAMMING [DSE 3254]

Marks: 50 **Duration: 180 mins.**

A

Answer all the questions.

Instructions to Candidates: Answer ALL questions Missing data may be suitably assumed
Note: Write code snippet as an example

1)	A)	A programmer needs to write a parallel program to find the minimum value of the given array of dimension N. He has preferred to do this using CUDA programming. Discuss with an example the performance consideration he has to take in order to do this task efficiently. Write a CUDA kernel function for the same.	(5)
	B)	With a neat diagram, contrast the various CUDA device memory types.	(3)
	C)	Assume that a CUDA device allows up to 12 blocks and 2,500 threads per SM, whichever becomes a limitation first. Furthermore, it allows up to 256 threads in each block. For matrix-matrix multiplication task, among the following (8x8, 12x12, 14x14) which is the best configuration of block dimension. Justify your answer.	(2)
2)	A)	A programmer has written an OpenMP program which uses nested parallelism. However, he has failed to realize unique threadIDs in his program. Discuss this limitation of nested parallelism with an example and provide the solution for the same.	(5)
	B)	With suitable example differentiate between *copy private* and *thread private* clause.	(3)
	C)	A programmer has written an OpenMP program to perform vector addition. He must decide whether he wants to go for parallelism depending on the length of the vector. Discuss and explain the OpenMP clause that need to be used.	(2)
3)	A)	Explain with suitable code snippet, various OpenMP loop optimization techniques in detail.	(5)
	B)	Compare the various work-sharing constructs among the threads in an OpenMP program.	(3)
	C)	Differentiate between Ordered clause and Ordered construct.	(2)
4)	A)	Explain with diagram and example the following Collective communication function of MPI i. MPI_GATHER ii. MPI_ALLREDUCE	(5)
	B)	Write an MPI program using collective communication routines to replace all even elements of array A to 1 and replace all odd elements to 0 of size N. Display the resultant array A. Assume N is	(3)

evenly divisible by number of process.

C) Contrast between MPI_Bsend and MPI_Ssend with suitable example. (2)

5) Write an efficient CUDA kernel function and kernel launch to perform element wise multiplication of two gray scale images of dimension (126 x 86). Consider that a thread block can have 256 threads in total. (5)

A)

B) With respect to parallel programming explain the following terms: (3)

i. Parallel execution time

ii. Efficiency

A parallel program P spends 73% of its serial time executing a loop that can be completely parallelized. Assuming that this part of the code (loop) is parallelized, what is the maximum speedup and efficiency that can be achieved by the same program given that it runs on an octa-core system?

C) With an example, describe the concept of *performance cliff* in CUDA. (2)

-----End-----

Parallel Programming (DSE3254)

Makeup exam

Type: DES

Q1. Elaborate the various synchronization constructs in OpenMP. (5)

Q2. Explain with an example the organization of threads in CUDA (3)

Q3. Assume that a CUDA device allows up to 8 blocks and 2,048 threads per SM, whichever becomes a limitation first. Furthermore, it allows up to 768 threads in each block. For matrix matrix multiplication task, among the following (14x14, 16x16, 32x32) which is the best configuration of block dimension. Justify your answer. (2)

Q4. With an example of matrix-matrix multiplication, discuss how we can reduce global memory traffic in CUDA programming. (5)

Q5. Discuss with an example how register and shared memory can be a limiting factor for parallelism in CUDA. (3)

Q6. Illustrate the favourable versus unfavourable memory access patterns for matrix element access in CUDA programming. (2)

Q7. Write an OpenMP program to perform matrix-matrix multiplication. Demonstrate data-level and task-level parallelism. (5)

Q8. With suitable example, contrast the various *schedule* clauses in OpenMP. (3)

Q9. Describe the scenario where relaxed consistency model is used in OpenMP. (2)

Q10. A programmer needs to write a parallel program to find the maximum value of the given array of dimension N. He has preferred to do this using CUDA. Discuss with an example the performance consideration he has to take in order to do this task efficiently. Write a CUDA kernel function for the same. (5)

Q11. In detail, explain the various shared memory systems. (3)

Q12. For finding the sum of N numbers, explain the usage of Reduction clause in OpenMP (2)

Q13. Explain with a neat diagram and example, the following collective communicati functions of MPI:

a) MPI_ALLREDUCE
b) MPI_SCATTER .

. (5)

Q14. Write an MPI program to replace all even elements of array A to 1 and replace all odd numbers in root process. Display the resultant array A, count of all odd and even number in the root process (3)

Q15. Differentiate between MPI_GATHER and MPI_BCAST. (2)