Lab Manual

of

High Performance Computing

(20DS509P)

By

Vaibhav Raval
23MDS010

DEPT. OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF TECHNOLOGY
PANDIT DEENDAYAL ENERGY UNIVERSITY
GANDHINAGAR, GUJARAT, INDIA
JANUARY-MAY, 2024

Index

Sr. No.	Problem Statement	Date	Sign.
1.	Write a program of matrix multiplication to demonstrate the performance enhancement done by parallelizing the code through Open MP threads. • Analyze the speedup and efficiency of the parallelized code. • Vary the size of your matrices from 5, 50, 100, 500, 750, 1000, and 2000 and measure the runtime with one thread. • For each matrix size, change the number of threads from 2, 4, 8, 10, 15, and 20 and plot the speedup versus the number of threads. Compute the efficiency. • Display a visualization of performance comparison between serial, parallel and NumPY code. • Explain whether or not the scaling behavior is as expected.	17/1/2024	
2.	Write a program for Leibniz series for PI calculation to demonstrate the performance enhancement done by parallelizing the code through Open MP work-sharing of loops. Display a visualization of performance comparison between serial and parallel, a visual analysis of delay/speedup with the help of varying thread counts and maximum terms in the series for Pi value calculation. • Implement the code with different thread count and different maximum number of terms to be calculated for the series such as thread count 10, 20 and terms 100, 1000, 10000, 1000000. • Display a visualization of performance comparison between serial and parallel, a visual analysis of delay/speedup with the help of varying thread counts and maximum terms in the series for Pi value calculation.	24/1/2024	
3.	Implement Producer-Consumer problem (PCP). Analyze the significance of semaphore, mutex, bounded buffer, producer thread, and consumer thread using the code available on Producer-Consumer Problem in Python - AskPython. Demonstrate how PCP occurs for an application of your choice.	29/1/2024	
4.	Write a program to generate and print Fibonacci series, one thread must generate the series up to number and other thread must print them. Ensure proper synchronization.	31/1/2024	
5.	Consider a scenario where a person visits a supermarket for shopping. S/He purchases various items in different sections such as clothing, grocery, utensils. Write an OpenMP program to process the bill parallelly	31/1/2024	

	in each section and display the final amount to be paid by the customer. Analyze the time take by sequential and parallel processing.		
6.	Implement the following programs of OpenMPI • Print "Welcome to PDPU from process (processno_totalprocesses)". • Apply denoising algorithm to a set of n images with 4 processes. (n=4, 8). • Analyze time taken by serial and openMPI processes. • Try for 100 or more number of images.	7/2/2024	
7.	1. Write a program to implement arithmetic calculations using MPI processes. 2. Write a program with different processes to apply following functions to an image in parallel. • Read an image. • Convert above RGB image to grayscale. • Find edges in the image. • Show the original image.	14/2/2024	
8.	Write a program to pass message from one process to another and print output. • In synchronous communication • In asynchronous communication. Show using overlapping of task in non-blocking mode.	14/2/2024	
9.	Calculate Pi value using openMPI send and receive messages for atleast 35-40 terms. Try the below mentioned commands, explain their task in one line and paste the output for each of them • Change the value on n as 2, 4, 8, 16. • Analyze the performance improvement using number of processes.	14/2/2024	
10.	Write a program to show collective communication by taking suitable example such that computing average of n numbers or computing sum or product of two matrices: • Bcast function • Scatter function • Gather function	14/2/2024	
11.	1. Describe Canon's Matrix Multiplication algorithm. 2. Implement Canon's Matrix Multiplication using collective communication. 3. Analyze the efficiency of the code.	19/2/2024	
12.	1. Write about derived data types used in MPI programming. 2. Steps to create and use derived data types. 3. Write its uses.	19/3/2024	

	4. Implement communication of derived data using one suitable example.			
13.	lshw (List Hardware) lsusb (List USB Devices) lspci (List PCI Devices) lsblk (List Block Devices) lscpu (List CPU) df (Disk Free) dmidecode (DMI Table Decode) ip a (IP Addresse)	top htop nvidia-smi lstopo perf numactl sar		
	For the given Python scripts that queries the CPU usage on a Linux-based system, understand the same and note the output for your device.		11/3/2024	
14.	Image Gray scaling Presentation		20/3/2024	
15.	Perform the following Image Processing Operations using the given images: • Image Blurring • Image Thresholding • Histogram based image analysis • Image Filtering/Denoising • Image Gray scaling		27/3/2024	
16.	Empirically understand and document the answers to the following: • What is CUDA? • What is the prerequisite for learning CUDA? • What are the languages that support CUDA? • What do you mean by a CUDA ready architecture? • How CUDA works? • What are the benefits and limitations of CUDA programming? • Understand and explain the CUDA program structure with an example. • Explain CUDA thread organization • Install and try CUDA sample program and explain the same. (installation steps)		1/4/2024	
17.	Implement following CUDA programs: 1. To print hello message on the screen using kernal function 2. To add two vectors of size 100 and 20000 and analyze the performance comparison between cpu and gpu processing 3. To multply two matrix of size 20 X 20 and 1024 X 1024 analyze the performance comparison between cpu and gpu processing 4. To obtain CUDA device information and print the output		15/4/2024	
18.	Implement the following Image Processing operations in sequential and parallel using CUDA Programming. 1. Gaussian Blur		22/4/2024	

	• Describe Gaussian Blur in brief. • Where parallelism can be inserted? • Analyze the performance in serial and parallel model. 2. FFT- Fast Fourier Transform • Describe FFT in brief. • Where parallelism can be inserted? • Analyze the performance in serial and parallel model.		
19.	Final Learning Synopsis Submission		