FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi

Lab Duration: 3 hrs. CS417 Parallel Processing Lab Marks: 10

Lab No: 06 Instructor: Ms. Anmol Sanaullah Khan Dated: 18/10/2024

Before performing tasks, keep in mind the following rules:

1. **CHEATING IS NOT ALLOWED. Looking at someone’s else screen is also cheating.**
2. **Mobile phone and internet usage are not allowed.**
3. **If you have any queries related to the task, you can ask instructors only. Never talk to each other until you are allowed.**
4. **Do not answer any query until you are asked.**
5. **Perform all the tasks.**
6. **Avoiding any of the above rules will lead to marks deduction.**

**Task 1 – Using Sections for Text Analysis**

Implement a program that performs text analysis on a given input string to compute the number of words, sentences, and paragraphs concurrently using OpenMP sections.

**Requirements:**

Use OpenMP sections to manage three separate operations: counting words, counting sentences, and counting paragraphs.

The main program should:

* Initialize a string with sample text.
* Use the sections construct to manage different computations.

Print the results of each analysis after all computations are completed.

**Expected Outcome:**

Text: "This is a sentence. This is another one! Here is a new paragraph.\n\nThis is a new line."

Word Count: 15

Sentence Count: 3

Paragraph Count: 2

**Task 2 – Calculations with OpenMP**

Implement a program that processes an array of integers to compute the sum, product, and maximum value concurrently.

**Requirements:**

* Use OpenMP to implement sections and tasks.
* The main program should:
* Initialize an array of integers.
* Use the sections construct to manage three separate operations: computing the sum, the product, and the maximum value.
* Within the sum and product calculations, use tasks to split the workload into smaller subtasks.
* Print the results of each operation after all computations are completed.

**Expected Output:**

Array: [1, 2, 3, 4, 5]

Sum: 15

Product: 120

Maximum: 5

**Hint:**

The sum and product will be calculated using tasks, while the maximum value will be found using a section.

Task 3 – Fibonacci Series Calculation

Implement a parallel program to calculate the Fibonacci series up to a user-defined number nnn. Each Fibonacci calculation should be computed using a recursive function, where the calculations are performed in parallel using OpenMP tasks.

Requirements:

* Use OpenMP to create tasks for each Fibonacci calculation.
* Implement a recursive function to compute Fibonacci numbers.
* The main program should prompt the user for the desired n value and display the Fibonacci series up to n.
* Handle input validation to ensure n is a positive integer.
* Ensure that all computed Fibonacci numbers are displayed in order.

Hint:

Enter a positive integer: 10

Fibonacci series up to 10:

1, 1, 2, 3, 5, 8

PRACTICE

Home Task – Linked List Traversal

Implement a parallel program to traverse a linked list and perform an operation (e.g., print node values) in parallel using OpenMP tasks.

Requirements:

* Define a linked list structure with a node containing an integer value and a pointer to the next node.
* Populate the linked list with sample data.
* Use OpenMP to create tasks for traversing the linked list.
* Each node's value should be printed by a separate task.
* Ensure that the linked list is traversed in order, maintaining the correct sequence of node printing.