



Lab2 Threads

Objectives:

1. Introducing threads concepts and POSIX threads library.
2. Implementing popular algorithms as multi-threaded ones.

Problem Statement:

It is required to implement two popular algorithms as multi-threaded ones. They are as follows:

1. Matrix Multiplication

It is required to implement two variations of this algorithm:

- a) The computation of each element of the output matrix happens in a thread.
- b) The computation of each row of the output matrix happens in a thread.

For both variations, you have to compute the elapsed time for each of them, compare them and justify your answer.

The program should read two input matrices in a certain format as mentioned in the “matrix-readme.txt” file associated with project (Output format is also mentioned).

2. Merge Sort

Merge sort is an $O(n \log n)$ comparison-based sorting algorithm. It is a divide and conquer algorithm. Conceptually, a merge sort works as follows:

- a) If the list is of length 0 or 1, then it is already sorted. Otherwise:
- b) Divide the unsorted list into two sub-lists of about half the size.
- c) Sort each sub-list recursively by re-applying the merge sort.
- d) Merge the two sub-lists back into one sorted list.

So you are required to implement it using Pthreads. Each time the list is divided; two threads are created to do merge-sort on each half separately. This step is repeated recursively until each sub-list has only one element.

The program should read two input matrices in a certain format as mentioned in the “merge-sort-readme.txt” file associated with project. When the program finishes, it should print out the sorted array.

Hint:

- Check pthread_join function.

Deliverables:

- Complete source code, commented thoroughly and clearly.
- Object Code.
- A report that includes:
 - A description of the overall organization of your code and the major functions.
 - Sample runs and screenshots.

Notes:

- Languages used: C/C++.
 - Operating System: Linux
 - Students will work individually.
 - *Revise the academic integrity note found on the class web page.*
-