Assignment HPC 3

Title:-
Parallel sorting Algorithms.

Problem statement:
for a bubble sort and merge sort based on existing sequential algorithms design and implement parallel algorithms utilizing all resources available

Objective:-
- To understand concept of Bubble sort and merge sort based on sequential algorithm.
- To understand concept of parallel algorithm
- To compare performance by varying number to processors used and also with sequential algorithm.

Outcomes:-
After successfully completing this assignment we can,
- Display result for parallel bubble and merge sort
- Analyze performance by varying number of processors used and also with sequential algorithm.

Software and Hardware requirement.
OS ; linux or windows.
openmp configuration.

Teacher Signature

Theory :-

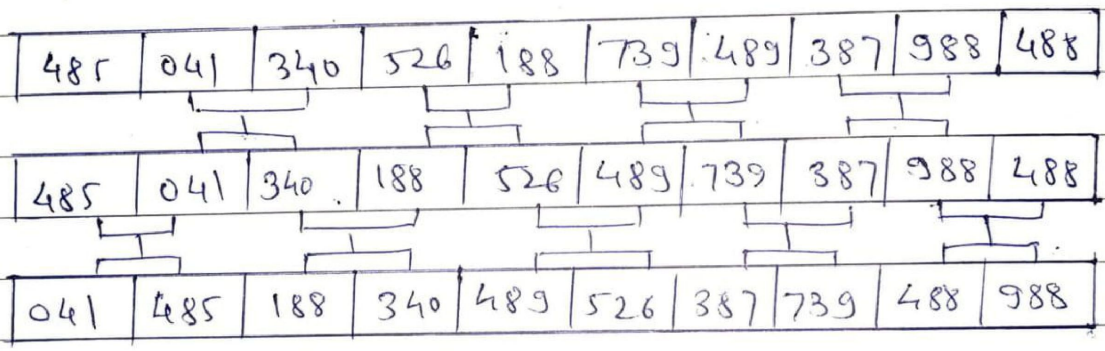
* Bubble sort algorithm

1. Sequential bubble sort algorithm

One of straight forword Sorting method.

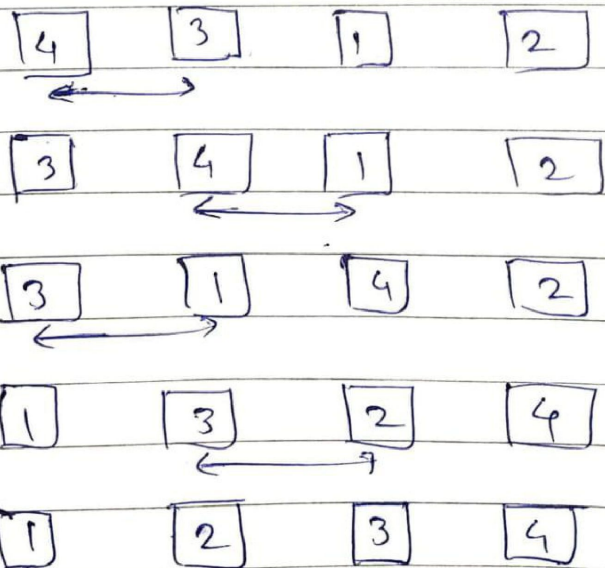
- cycles through the list.
- Compares consecutive elements and swaps them if necessary.
- stops when no more out of order pair
  - Slow and ineffcient
  - Average performance - $O(n)^2$

2. parallel bubble sort :- Compare all pair in list in parall



parallel bubble sort complexity.

Do (n-1) comparision For each iteration hence O(n)



Teacher Signature

* Merge sort algorithm.

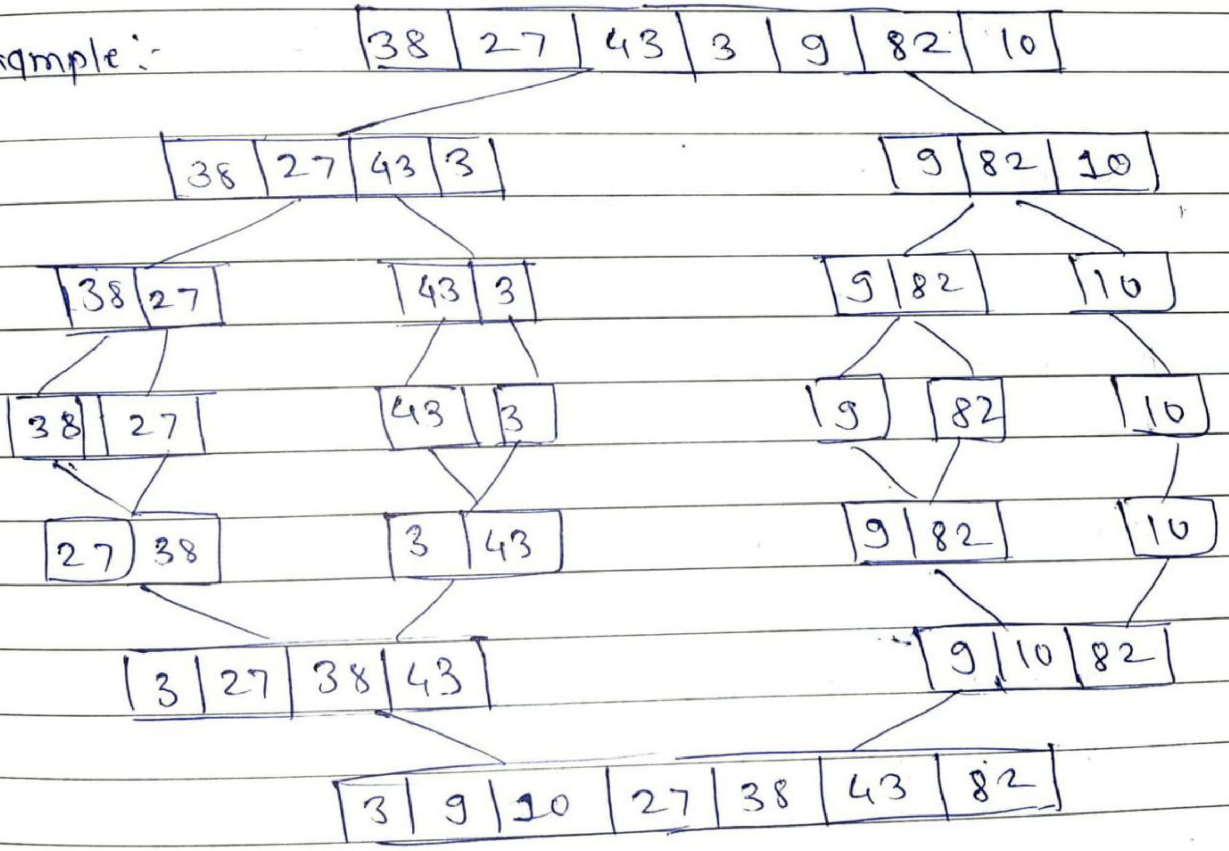
Divide and conquer ~~et~~ approch

- Divide problem into subproblems
- Division usually done through recursion
- Solutions from subproblems then combined to give functional solution to original problem

Collect sorted list onto one processor

- Merges element as they come together
- Simple tree structure.
- parallism is limited when near the root.

Example:-



Complexity

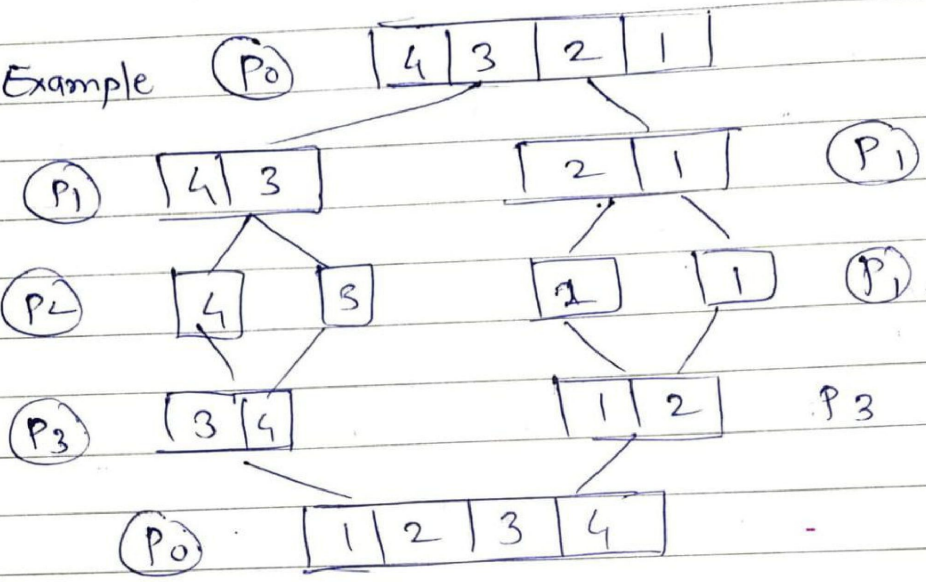
$T(n) = O(n \log n)$

Teacher Signature _____

parallel merge sort

- parallelize problems
- Max parallelism is achieved with one processor

Example



Complexity :- O(log (n))

Testcase : * Bubble sort -

Array element : 6

elements :- 6 5 4 3 2 1

Sorted array :- 1 2 3 4 5 6

Time required ~~of~~ for parallel : 0.003000

* merge sort

Array element : 6

elements : 6 5 4 3 2 1

Sorted Array : 1 2 3 4 5 6

Teacher Signature

Time required for parallel : 0.000399995

Conclusion:- After succesful implementation for bubble sort and merge sort using parallel approch calculated time required for parallel bubble sort and merge sort.