**Parallel Algorithms**

**Assignment 3**

**Group: 5**

**Venkata Govardhan Goli (7)**

**Suhas Sai Raparthi (22)**

**Mohammed Belhadj Larbi.**

1. Design a parallel sorting technique and implement it in spark.

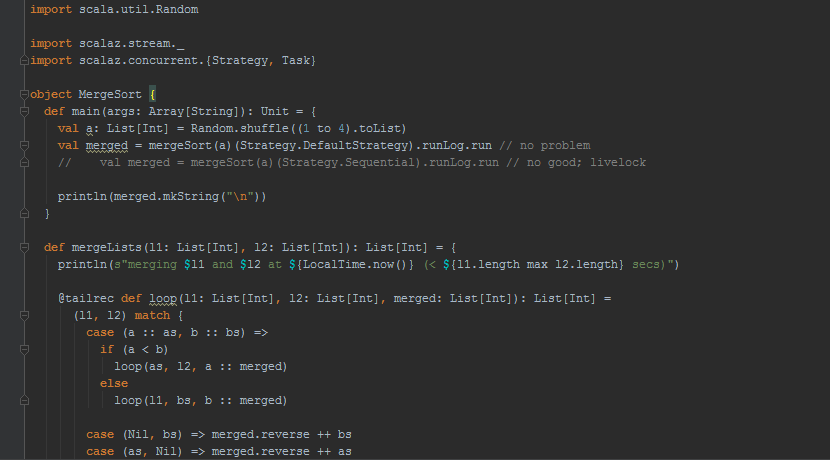
Submit the following

* 1. Code through Github
  2. Report with Algorithm, Sample Input and Output, Complexity and Time Performance.
  3. Bonus for comparative evaluation with other techniques

Github: <https://github.com/venkatagovardhan/Parallel-algorithms.git>

We want to implement the merge sort in parallel manner using the scala coding.

Merge sort in sequential manner.



In sequential algorithm

If the total numbers are >1 the total numbers are divided into two halves.

Repeat the above step until only one number is remained.

Later combine the numbers by sorting them.

This continues until all the numbers are sorted.

The complexity of the sequential merge sort is O(nlgn).

In parallel manner

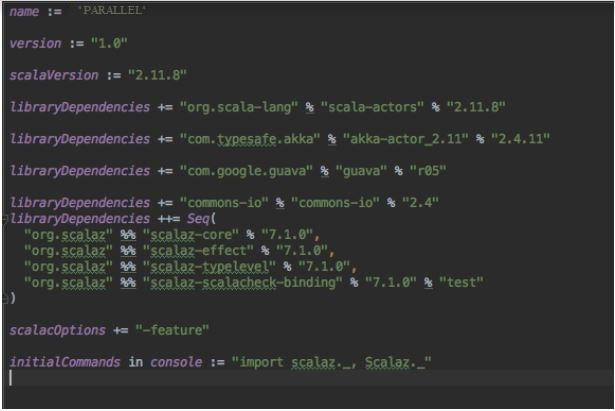
In parallel merge sort the small sets are given as the inputs to the processors.

These processors sort them and merge them into the sorted manner.

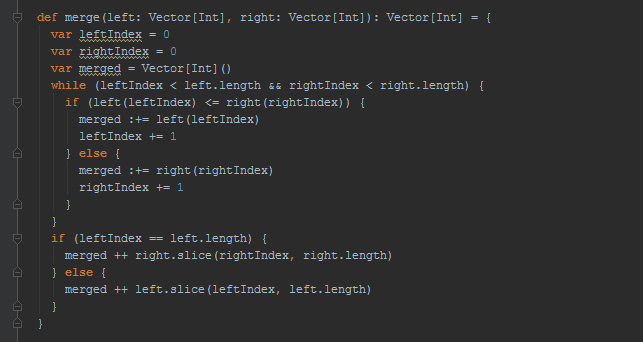
The number of processors that are required at the starting level are depend on the total number of items in the sorting list.

If the items are divided into 2^k items with N number of processors, then complexity is O(N/k\*lg(N/k)).

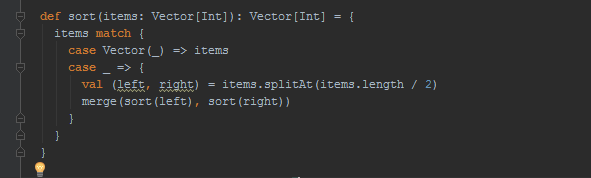
For parallel algorithm following dependencies in scala sbt in intellij



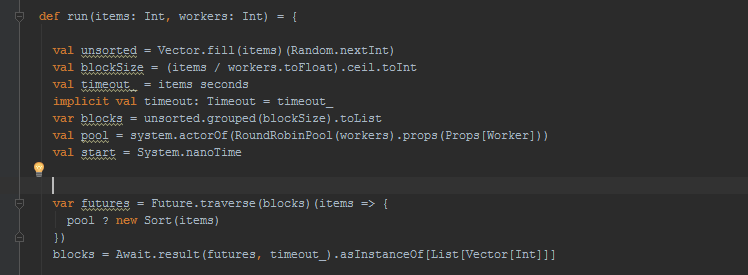
Merge the right side and left side into one.



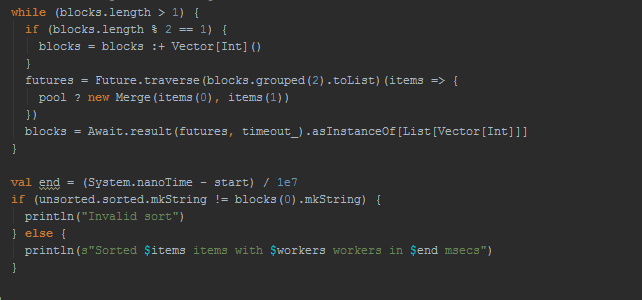
This sorting sort the items and then merge the items in left and right.



First split the items into partitions and then sort the items. Each partition is processed by separate actor.



Later merge the results until it all becomes as one item.



**Quick sort:**

In quick sort pivot number is selected and the list is divided into two parts. One with the elements less the pivot and other with elements greater than pivot.

This process continues until every element comes in a sorted order.

When compared it with the quick sort merge sort is little better. Both quick sort and merge sort does not use the processors efficiently.

The complexity of

Parallel Quick sort is- O(n)

Parallel merge sort is O(n).