

Assignment #4

General Instructions:

- The aim of the assignment is to get familiar with Erlang (<https://www.erlang.org/>).
- The assignment has to be done using Erlang only.
- Strict actions would be taken against anyone found involved in any kind of plagiarism either from the internet or from other students.
- Submission instructions given at the end of the assignment must be strictly followed. Since the evaluation will be automated, any deviation will result in loss of marks.

Submission Instructions:

- The program will be given two arguments, path to an input file and an output file from which your program will obtain the input and output the result into respectively.
- The submission is expected to be a <RollNumber>.zip file containing a directory with the same name as your roll number that holds the following files:
 - A program file for each of the mentioned problems with the name: <RollNumber>_<ProblemNumber>.erl. (eg. 20161215_1.erl)
 - A brief report describing and analyzing your solution as: README.md

(1) Write a program to pass an integer token value around all processes in a ring-like fashion, and make sure that it does not have a deadlock.

Input Format:

Input contains two space-separated integers denoting the number of processes and the token value respectively.

Output Format:

Check sample output.

Sample Input:

5 1

Sample Output:

Process 1 received token 1 from process 0.
Process 2 received token 1 from process 1.
Process 3 received token 1 from process 2.
Process 4 received token 1 from process 3.
Process 0 received token 1 from process 4.

(2) **Parallel Merge Sort:** Given an array of numbers, your task is to return the array in sorted order by implementing parallel merge sort.

Input Format:

The first and the only line of input contains an array of integers.

Output Format:

Output your answer as a space-separated array.

Sample Input:

7 -1 4 9 -8 0 6

Sample Output:

-8 -1 0 4 6 7 9