Distributed Systems Assignment 2

Erlang

Due: February 18, 11:55PM

1 Overview

The aim of the assignment is to get familiar with Erlang programming language (https://www.erlang.org/).

2 Important Points

For this assignment, you will be coding solutions to different problems in Erlang.

- **Installation Guide:** https://medium.com/@brucifi/erlang-quick-install-a3b7fd96947f
- **Introduction to Erlang:** https://www.ibm.com/developerworks/library/os-erlang1/index.html
- **Erlang Documentation:** https://www.erlang.org/docs
- **Command for Erlang man pages:** erl -man <name>
- Commands to execute your code:
  - **Compile:** erlc <roll_number>_1.erl
  - **Execute:** erl -noshell -s <roll_number>_1 main <input_file> <output_file> -s init stop
- Make sure that the module name and the name of the exported function are correct. For module name, put the roll number as a string, for example, module('2019XXXX_1') .

3 Problems

Your 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. Maximum number of processes will be 50.

Problem 1 (10 points)

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

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

Output

Check sample output.

Sample Test Cases

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.
```

Problem 2 (40 points)

Given a weighted graph and a source vertex in the graph, find the shortest paths from source to all vertices in the graph. You can use any algorithm to solve this problem.

Input

The first line of input contains number of processes **P**.
The second line of input contains **N** and **M** representing the number of vertices and edges in the graph respectively. The vertices will be numbered from **1** to **N**.
Each of next **M** lines contains the triplet **X**, **Y** and **W** that represents the two edge points and the weight of an edge.
Next line contains **S**, the source vertex.

Output

Print the shortest distance of each vertex from the source vertex on a new line, starting from the lowest vertex as follows:
<Vertex No.> <Distance from Source>

Constraints

$1 \leq P \leq N$

Sample Test Cases

Sample Input

```
3
4 5
1 2 1
1 3 1
1 4 3
2 4 1
3 4 1
1
```

Sample Output

```
1 0
2 1
3 1
4 2
```

4 Submission Instructions

Your 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**
- A brief report describing and analyzing your solution as: **README.md**

Since your codes will be evaluated automatically, you have to follow the output format strictly.

NOTE: Strict actions would be taken against anyone found involved in any kind of plagiarism either from the internet or from other students. If we find any of the codes implementing the question in serial manner or using other algorithms than mentioned in the question, then it will result in serious penalties.