Operating Systems

Laboratory 12

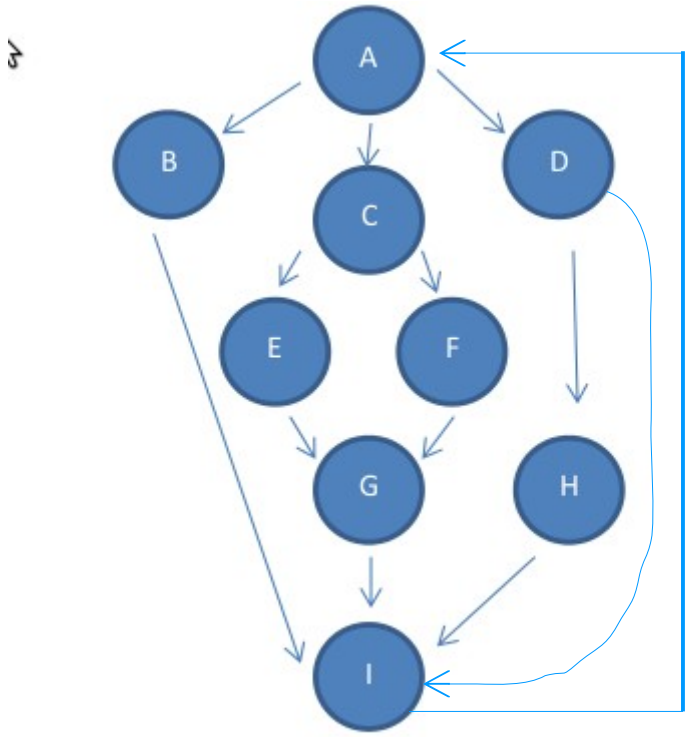
Learning goals:
In this laboratory you will improve your precedence graph understanding. You will also improve your knowledge about AWK arrays usage. Finally you will be familiar with SED tool.

Exercise 1

Make the following precedence graph using the minimum number of pipes in order to synchronize:

1. processes ;
2. cyclic processes;

Each circle represents a separate process.



Hint
The precedence graph described above is very similar to the one described in the laboratory 11, just some modification, to the previous program you wrote, are needed to complete this exercise.

Exercise 2

A program, useful to organize projects, stores its results on file with the following line format:

- “project name” “execution time1” “execution time2” “execution time3”

File format example:

progA 12.34 26.45 123.99
progB 32.45 16.45 23.23
ex001 56.34 6.45 343.99

pdtsw 112.84 265.45 56.82

Write an **AWK script** able to:

1. read two files that have the format described above(one of them has to be the default input)
2. display on standard output the list of all projects stored in the **two files** in the following format:
   - "project name" "execution time1" "execution time2" "execution time3" "sum" "string"

The "sum" field contains the sum of the 3 executions time, the field "string" will be equal to the string "ONE" if the project is present only in one file, and "BEST" if it is present in both.
In the second case("BEST") the executions time correspond to the project which has the lowest sum.

Example:

FileA
progA 12.34 26.45 123.99
progB 32.45 16.45 23.23
ex001 56.34 6.45 343.99

FileB
pdtsw 112.84 265.45 56.82
progB 15.45 20.45 60.23
progA 6.34 13.45 60.99

Result
progA 6.34 13.45 60.99 **80.78 BEST**
progB 32.45 16.45 23.23 **72.13 BEST**
pdtsw 112.84 265.45 56.82 435.11 ONE
ex001 56.34 6.45 343.99 406.78 ONE

Keep in mind that the projects present in the two files are not stored in the same order.

Exercise 3

Write a SED script that:

- Replaces the first five lines of a input file with a single line containing the string "HEADER";
- Add three spaces at the beginning of a line whose line number is between 6 and the end of file;
- Delete all the lines starting or ending with a digit;

(Each command has to be applied separately).

Summary
At the end of this laboratory activity you should know how to synchronize cyclic process using the minimum number of semaphore. You should also know how to use conditions and arrays in AWK and finally you should be able to perform simple tasks using SED.