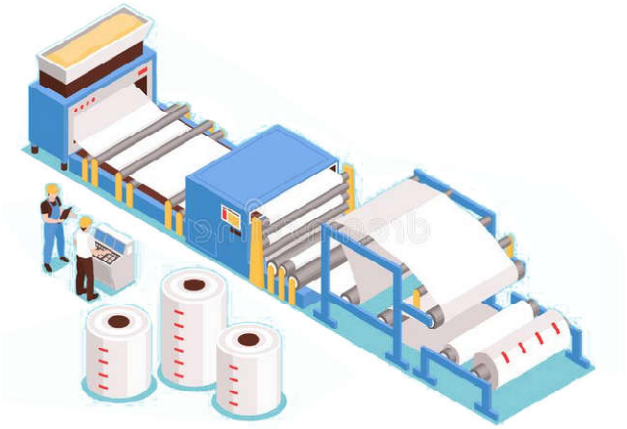
COMP301 - OPERATING SYSTEMS

Fall 2022
ASSIGNMENT - 2
DUE: 18th January'2023, 11:59PM

Task

- Write two C programs for Linux that use the IPC (inter process communication) signals.
- Both programs have to run in different shell (terminal) windows on the same system.
- The first program is called "papermachine.c" and the second one is called "paperbox.c".
- papermachine.c represents a worker who is responsible for making paper from wood pulp.
- paperboxes.c represents a worker who is making paper boxes using paper.



papermachine.c

paperboxes.c

- Both the processes are able to run in parallel. The product of papermachine.c (paper) is delivered to paperboxes.c, and paperboxes.c needs the paper to work.
- Therefore, if the paperboxes runs out of paper, it sends a signal to the papermachine to provide it with more paper.
- Paper is prepared in the following way:
    - A fixed small delay (e.g., 0.3 seconds) is needed to prepare the paper.
    - The machine does not deliver the paper unless a total of 10 yards of paper are completed.
    - A variable goes from 0 to 10 when the paper is fully prepared.
    - When paper is delivered, this variable goes to 0 and starts over.

  - The paperboxes goes into waiting if the paper is being prepared, and his batch is fully converted to boxes.

- Paperboxes works in the following way
  - A fixed small delay (e.g., 0.4 seconds) is needed to make one box.
  - One batch of paper can only be used to make 5 boxes.
  - A variable goes from 0 to 5 when a batch is complete.
  - When the batch is complete, this variable goes back to 0 and starts over.
  - The papermachine goes into waiting if the boxes are being made, and paper is new paper is not needed by paperboxes.
- Use the kill() system call and User-Signals to simulate the behavior mentioned above.
- Messages/symbols on each terminal should be printed at least in case of the following events.
  - The paperboxes goes into waiting due to no paper.
  - The paperboxes starts making boxes.
  - Each time paperboxes makes one box.
  - The paperboxes finishes making boxes.
  - The papermachine goes into waiting due to boxes being made.
  - The papermachine starts making paper.
  - Each time the papermachine makes one yard of paper.
  - The papermachine finishes making paper.
- You can improve the appearance and display of the simulator to as much extent as possible for you.

SUBMISSION DETAILS

1. File 1: papermachine.c (the program must compile without any errors and work as described above)
2. File 2: paperboxes.c (the program must compile without any errors and work as described above)
3. Upload the 2 files to Moodle.