Virtual Lab Experiments for Operating Systems

2025-02-05

Contents

List of Figures

1 Virtual Lab Name and Discipline

Name of Lab: Virtual Lab in Operating Systems
Discipline: Computer Science and Engineering
Target group: UG 2nd Year (4th semester)

2 Relevant AICTE Course

Operating Systems is a core computer science course in the AICTE model curriculum PCC-CS403, page 342.

3 List of Experiments

The experiments being proposed are listed below:

No.	Topic (AICTE Module)	Experiment
1.	Processes (Module 2)	Process State and Transitions, Context Switch

4 Related Virtual Labs available on the Web

To the best of the author's knowledge, the lab is also not currently available on `https://vlab.co.in` website or any other site on the web.

5 Pedagogical Value of Experiments

The Virtual Labs for Operating Systems complement the theory (textbook) approach and the programming approach to understanding operating systems. The focus of these experiments is to build a collection of small *simulations* in which the student may either (a) watch the execution of an algorithm or a policy, or (b) drive the experiment by using controls (buttons) to achieve the result of simulating a given algorithm or operating system policy.
In the next few subsections, we sketch the need and structure of the experiments we wish to build. More details of the structure of the experiments will become apparent once their development commences.

5.1 Processes and Context Switching

5.1.1 Topic description

An operating system is designed to execute programs. A process is a program in execution. Running processes is the fundamental purpose of an Operating System. A process is defined by the values of its registers and program counter, collectively called the context of the process. A process may be either running inside the CPU, or may

be waiting to be run by the CPU, or terminated. A process does not continuously occupy a CPU; instead it runs for some time and relinquishes the CPU and runs again, etc. This phenomenon is called context switching. A context switch happens, for example, when the process is waiting on a lock, or about to perform an IO operation or a system call, or when it is preempted by the CPU to let another process run as part of the CPU's scheduling policy.

5.1.2 Need for virtual experiment

Processes and context switching involves multiple processes in multiple states, along with the per-process data structures like Program Control Block, kernel stacks, etc. Most textbooks have a verbose description of these structures and the actions that drive context switches. A virtual lab for processes and context switch would allow the student to identify the main actors, the actions involved in the dynamics, and the consequences of those actions on the states of the processes and the data structures involved. By manually manipulating and driving the simulation, the student would be able to recreate the steps involved in a context switch.

5.1.3 Experiment Description

- **Objective:** The objective is to simulate the life cycle of a process created in the execution of programs.
- **Visualisation (View):** The simulation will allow the student to view a collection of processes, and for each process, its state, process control block and other data structures.
- **Controls:** The controls available drive the creation of a process and its associated data structures, the switching of contexts for a process from a running to waiting state and restoration of data when the process runs again. Through the controls, the student is expected to 'drive', for a few steps, the process lifecycle and context switching that is orchestrated by the operating system.
- **Validation:** The sequence of student actions and states captured in a log may be validated to ensure that the per-process data structures are correctly manipulated by the student during context switching and process execution.

A schematic of the experiment's visual interface is shown in Figure 1.

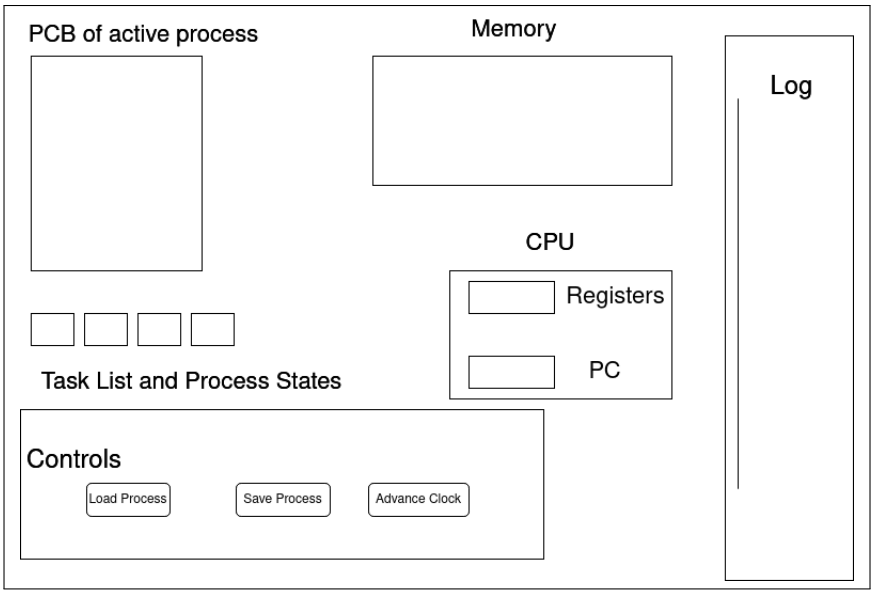

Figure 1: Schematic visual interface the Process Life-cycle and context switching virtual experiment

6 Familiarity and experience with the subject matter

The author has previously built a Virtual lab for Linux. His area of research includes modular design of mutual exclusion algorithms included in the list of experiments and formal verification of system software. The author has taught several of the listed topics in the software and systems foundations course at IIITH for the last several years.

7 Technology

All the experiments will be developed as static web applications using Javascript (or Elm, which compiles to Javascript), HTML5 and CSS.