



## Tutorial: Process Control Blocks

1. What is a *process control block* (PCB)?



2. Is a PCB only found in a Linux or POSIX OS or is it found in all operating systems?



3. Give *ONE* other name for a PCB.



4. Give *ONE* name for the data structure that holds the PCBs.



5. Briefly define the term *program counter*.



6. What data is held in the *program counter*?



**Process Control Blocks:** In figure 1 on the following page we see three processes, P1, P2 and P3. There are seven moments in time where the main values of the PCBs of the three processes are shown. The time intervals are not equal; they are just where the scheduler changed the process. The size of all I/O instructions is *three bytes*. In each PCB I have shown the program counter value, and the process state. The process state can be running (i.e., executing on the CPU), ready (i.e., ready to run), and sleeping (i.e., blocked). There are two queues shown: a wait queue for processes that are blocked waiting for I/O, and a ready queue, for processes that are in the ready to run state.

1. State the technical term for what the scheduler does at each of the times  $t_0$ ,  $t_1$ ,  $t_2$ ,  $t_3$ ,  $t_4$ ,  $t_5$  and  $t_6$ .



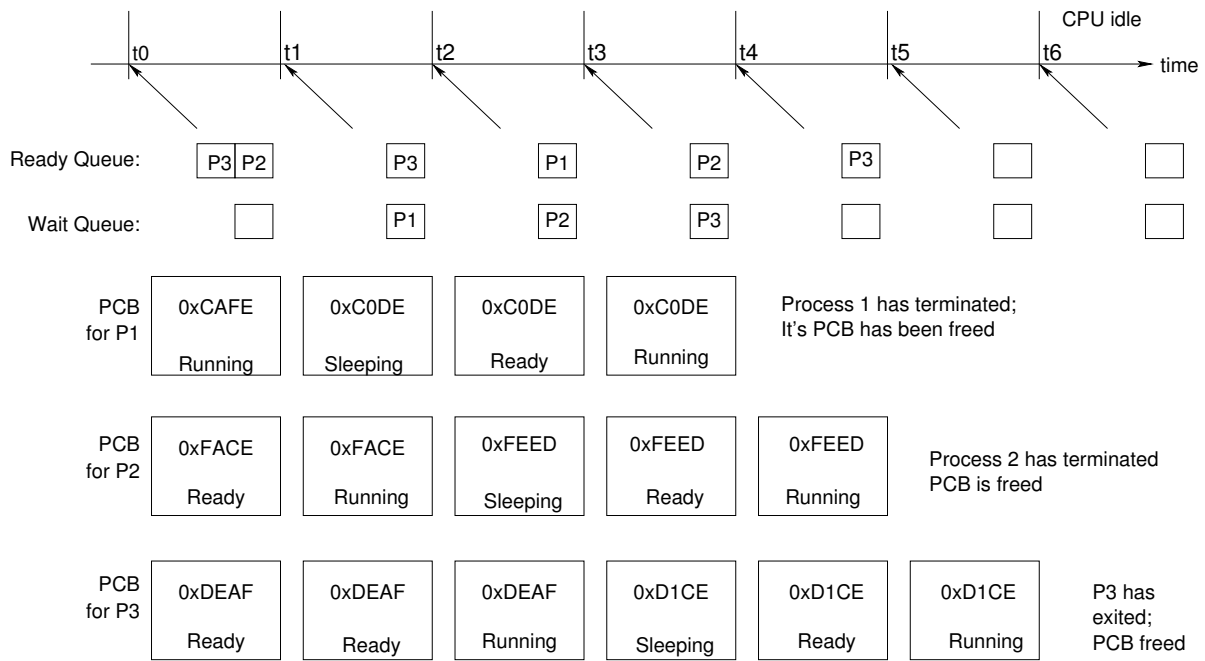


Figure 1: An example of three processes executing, showing their process control blocks.

2. Does process P1 perform I/O? If so, at what address is the I/O instruction?



3. Does process P2 perform I/O? If so, at what address is the I/O instruction?



4. Does process P3 perform I/O? If so, at what address is the I/O instruction?



5. List at least *FIVE* other kinds of data that will be stored in a PCB that is not shown

in figure 1 on the previous page.

