Help for simulation assignment.

Every process has two types of variables

1. Initial values that do not change, each process has different starting values
2. Current values and counters that change during its life

Example:

1. Priority, the initial priority of a process that it enters the ready queue with
2. Current priority, changed as the process ages in the ready queue, set to initial priority on first entering ready queue.

**Life of a process from creation till termination**

When the process is not in the wait queue it’s time in the CPU or doing I/O is kept track of via the total time in machine. So, if that starts at 1000, every time click, in the CPU or I/O subtract 1. Time waiting in wait queue is not part of the life of a process.

When a process has used all its time it is terminated.

**Life of a CPU bound process**

On entering the CPU Time in CPU is set to 0.

Every time click, it is incremented by 1 and compared to the Quantum and time before I/O

When it times out it is moved to the ready queue and the Total time in CPU is updated

The time process has been in ready queue is set to 0

Every time click, it is incremented by 1

When it leaves the ready queue for the CPU the Total time, smallest and longest times in ready queue are updated.

**Life of an I/O bound process**

On entering the CPU Time in CPU is set to 0.

Every time click, it is incremented by 1 and compared to the Quantum and time before I/O

When it needs to do I/O it is moved to the wait queue and the Total time in CPU is updated, The time waiting for I/O is set

Every time click, the Time waiting for I/O is changed and compared to time I/O takes

When the I/O has finished the process is moved to the ready queue

The time process has been in ready queue is set to 0

Every time click, it is incremented by 1

When it leaves the ready queue for the CPU the Total time, smallest and longest times in ready queue are updated.

**Priority Queue**

Each process time in ready queue is set to 0 on being put in queue and its current priority is set to the initial priority

Each time click, the time in ready queue for each process is incremented

Each time click, the time in ready queue for each process is compared to max wait time

Each process has its current priority adjusted as is approaches the max wait time

The queue is resorted by current priorities

**Wait for I/O list**

Should really be called wait for interrupt. When I/O is over it interrupts and the kernel moves the process to the ready queue.

A process may be put in the wait queue by the sleep( ) function or put to sleep waiting for an event like a signal, not just waiting for I/O.

In the simulation, it is a counter till wait is over.