CPU Scheduling

CS439: Principles of Computer Systems

Bringing It Together

- OS has three hats: What are they?
  - Processes help with one? two? three? of those hats
- OS protects itself and others through dual-mode execution
  - Which relates to processes how?
- So then we got to the illusionist part: CPU Scheduling
  - discussed the possible policies the scheduler may use to choose the next process (or thread!) to run
  - criteria we use to evaluate policies
    - throughput, turnaround time, response time, CPU utilization, waiting time (and one more! What was it?)
  - FIFO, Round Robin, SJF, Multilevel Feedback Queues

Concept Map Possible Terms

PCB
blocked
turnaround time
mode switch
long-term scheduling
throughput
timer interrupt
knowing the future
predicting the future
short-term scheduling
starvation
ready

Round Robin
concurrency
waiting time
MLFQ
ready queue
CPU utilization
running
ideal CPU scheduler
scheduling quantum
scheduler
FCFS/FIFO
non-preemptive

priority
response time
overhead
fairness
SJF
time slice
I/O bound
preemptive
time sharing
context switch
multiprogramming
CPU bound

Boot Sequence:

UEFI/BIOS
boot program
initialization of system
idle loop
kernel
busy waiting
machine configuration

- Say “hi!”: ***Introduce yourself (name, and how you are).***
- Each participant share: ***If I had a pet dinosaur, I would...***

- Consider a process that calls fork(). Immediately after the call, will the parent or child execute first? Defend your answer.
- Who executes the scheduler? As in, who runs the scheduler code?

Something Different

Exit Ticket

- Name
- EID
- Two criteria for evaluating CPU schedulers are throughput and response time.
  - What algorithm optimizes for both?
  - Name an algorithm that optimizes for throughput (but not response time).
  - Name an algorithm that optimizes for response time (but not throughput).
  - What potential problem faces algorithms optimizing for throughput?