CS 1550: Scheduling

Jonathan Misurda
jmisurda@cs.pitt.edu
http://www.cs.pitt.edu/~jmisurda/

Multiprogramming

Single SPC
(CPU's point of view)

Multiple SPCs
(process point of view)

Process
A running program and its associated data

Life Cycle of a Process

Created
Ready
Blocked (waiting)
Running
Exit

Process Table

Process Table Entry

May be stored on stack

File management
Root directory
Working (current) directory
File descriptors
User ID
Group ID

Memory management
Pointers to text, data, stack or
Pointer to page table

Process management
Registers
Program counter
CPU status word
Stack pointer
Process state
Priority / scheduling parameters
Process ID
Parent process ID
Signals
Process start time
Total CPU usage
Thread
A stream of instructions and their associated state

Processes and Threads

Thread State

Per process items
- Address space
- Open files
- Child processes
- Signals & handlers
- Accounting info
- Global variables

Per thread items
- Program counter
- Registers
- Stack & stack pointer
- State

Threading

Multithreading in Action

Multithreaded Webserver

while(TRUE) {
    getNextRequest(&buf);
    handoffWork(&buf);
}

while(TRUE) {
    waitForWork(&buf);
    lookForPageInCache(&buf,&page);
    if(pageNotInCache(&page)) {
        readPageFromDisk(&buf,&page);
    }
    returnPage(&page);
}
User Threads vs. Kernel Threads

Scheduling
How to choose which of the Ready processes/threads gets to Run next

CPU Bound vs. I/O Bound

When to Schedule
- Process Creation
- Process Exit
- Blocked
- I/O Interrupt
- Clock Interrupts

Three-Level Scheduling

Fairness
Comparable processes get comparable service
Throughput
*Number of jobs completed per unit time*

**Turnaround Time**
*Time from job submission to job completion*

**Average Turnaround Time**
*Average of all turnaround times for a set of jobs*

**Batch Scheduling**
*Non-interactive jobs that can be run “overnight”*

**First Come, First Served**

**Shortest Job First (SJF)**

**Interactive scheduling**
*Impatient users waiting*
Round Robin Scheduling

Priority Scheduling

Other Scheduling Algorithms

• **Shortest Process Next**
  – *SJF applied to Interactive Systems*

• **Guaranteed Scheduling**
  – *N processes get 1/N of the CPU Time*

• **Lottery Scheduling**
  – *Give out tickets, pull one at random, winner runs*

• **Fair Share**
  – *N users get 1/N CPU time*

**Earliest Deadline First (EDF)**

*Real-time: How you do homework*

**Mechanism**

*The way something is done (e.g., an algorithm)*

**Policy**

*The rules a particular mechanism should follow (i.e., the parameters of an algorithm)*
Scheduling User Threads

Scheduling Kernel Threads