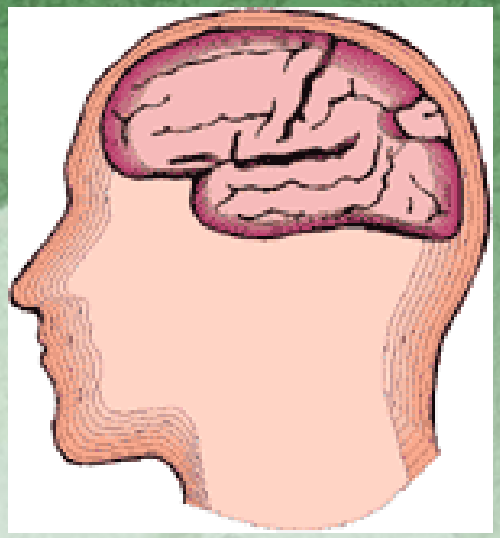
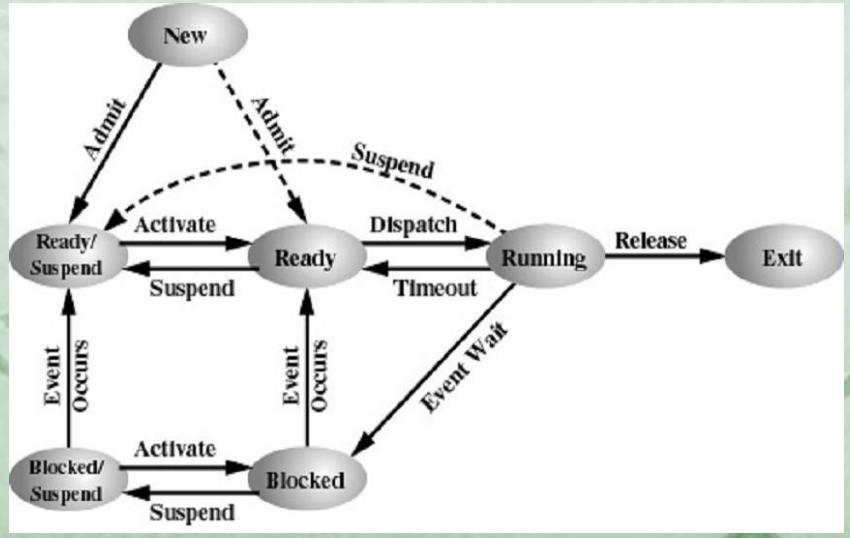


Operating Systems



Course Information

Description:

Operating Systems in theory and practice. Components in a system: scheduling and resource allocation; process management, multi-programming, multi-tasking; I/O control and file systems; mechanisms for client-server computing. Examples from typical operating systems. Prereq's: (CIS*2500 or CIS*2650).

Text Book: **Operating System Concepts, 9th Ed.**

Silberschatz, Galvin, Gagne,

John Wiley & Sons. INC

Topics:

- Structure of a process : address space, registers, program counter and the stack.
- Asynchronous activity : hardware and software interrupts, service routines.
- Multiprogramming : programming for concurrency, shared data, race conditions, critical sections and tools/techniques for multiprogramming such as mutual exclusion, semaphores and strict message passing.

Topics:

- Memory management: memory allocation, stack, heap, virtual memory.
- File systems: file organization and interfacing with secondary storage.
- The operating system kernel : internals and interfacing via system calls.
- Distributed computing: communication, client/server model, remote procedure calls.

Syllabus

- At the end of the course, the student should have a basic understanding of:
  - Design and implementation issues of operating systems
  - Detailed analysis of process, multi-threading, symmetric multiprocessing, and micro-kernels
  - Memory management techniques, including virtual memory
  - Various approaches to process scheduling
  - Operating system control of Input/Output
  - Operating system management of files

Prerequisites

- Knowledge of C, C++, Java
- Ability to complete large programming project
  - Understand existing framework
  - Design system solutions to particular problems
- Able to use Windows, Unix, Linux, MacOS, …

Prerequisites

- Linked List, Stack, Queues, Priority Queues, *etc.*
- OS is built on the top of queues
- Data Structures?

Assignment Requirements

- You are encouraged to discuss assignments with fellow students. However, you are **not** allowed to share code with any student.
- All assignments are submitted through the on-line Moodle.
- Each student should submit a brief report (2-3 paragraphs) that explains your algorithm, any assumptions you made, and the way to run your program.
- If your TA is unable to run/test your program, you should present a demo arranged by your TA.
- Programming language: c, c++, gcc
- Operating System: Unix, Linux, or Unix emulator (cygwin).

Organization and Grading

- This class consists of lectures, programming assignments, a mid-term exam and a final exam. Each has a weighted contribution to your final grade.
  - **Assignments:** 40%, where
    - Assignment 1: 10%
    - Assignment 2: 15%
    - Assignment 3: 15%
  - **Mid-term Exam:** 20%
  - **Final Exam:** 40%

Course Information

Course webpage can be found at:

http://moodle.socs.uoguelph.ca/

Enrollment Key: cis3110

Lecture Time: Mon. Wed. Fri. 11:30-12:20 AM, ALEX 200

Labs: Fri. 9:30-10:20 AM ROZH 105

Thur. 10:30-11:20 AM ROZH 105

Thur. 8:30 – 9:20 PM ROZH 105

Thur. 4:30 – 5:20 PM ROZH 105

Office Hour: Tuesdays 9:00-11:00 AM

Schedule

- Assignment 1 Due: Jan. 30, 2017
- Assignment 2 Due: Feb. 27, 2017
- Mid term exam: Mar. 1, 2017, in class room
- Assignment 3 Due: Mar. 27, 2017
- Final Exam: 7:00 – 9:00 PM, Apr. 13, 2017, Room TBA

Academic Honesty

- Academic honesty includes completing your own assignments and exams.

- Examples of academic dishonesty include sharing code for assignments with other students, turning in someone else's writing as your own report, and cheating on an exam.

Miscellaneous

- TA’s

  See Moodle for detailed information, such as

- Help sessions
- TA Office hours
- Linux vs Microsoft users (cygwin)
- Questions?

Self-test:

(1) Lifetime of a variable is defined by its

a) data type
b) storage class 
c) name
d) memory address

(2) If a local variable has the same name as a global variable, what will happen?

a) A compiler error gets generated.
b) Both variables will share the same memory location.
c) The local variable will supersede the global variable, *i.e.* hide it. 
d) Both are visible to the entire program.

(3) Given the declaration, which scanf would successfully read in a float value into x?

```
float  x, *y = &x, **z = &y;
```

a) `scanf("%f", &x);` 
b) `scanf("%f", *y);`
c) `scanf("%f", *z);` 
d) `scanf("%f", y);` 

(4) Given the following expression which of the following is not true?

s[3].x[0]->t

a) s is an array.
b) x[0] is a pointer to a structure.
c) t is a member of s[3].
d) x is a member of s[3].

(5) Text files are nice because they

a) are more efficient than binary files.
b) are portable among different platforms.

c) take up less space than binary files, typically.
d) can be accessed randomly.

(6) Given that x is an unsigned char,

x ^ 0xFF will always give

a) all zeros
b) all ones
c) x
d) none of the above