CSE381 Quez1 Study Guide

Chapter 2

Section 2.1, 2.2, 2.3, 2.4, 2.5, 2.6 and 2.7.

* The concept of process, process control block, process elements, the difference between process and program

A **process** is the execution of an application program with restricted rights; the process is the abstraction for protected execution provided by the operating system kernel.

A **PCB** is a data structure that stores all the information the operating system needs about a particular process: e.g., where it is stored in memory, where its executable image is on disk, which user asked it to start executing, and what privileges the process has.

* User mode, kernel mode, privileged instruction, memory protection

* The concept of interrupt, polling, system call, processor exception, the reasons to cause mode transfer

An **interrupt** is an asynchronous signal to the processor that some external event has occurred to require its handling.

A **system call** is any procedure provided by kernel that can be called from user level.

A **processor exception** is Hardware event caused by user program execution

User -> Kernel: Interrupts, Exceptions, System calls

Kernel -> User: New process, return from interrupt/exception/system call, Process context switch, User-level upcall

* Interrupt vector table, interrupt handler, interrupt handler, two stacks per process

**Interrupt vector Table** set up by OS kernel; pointers to code to run on different events

An **interrupt handler** is the term used for the procedure called by the kernel on an interrupt

**Interrupt masking** prevents interrupts from being delivered at inopportune times.

* Mode transfer implementation (section 2.5)

You need to understand every details regarding mode transfer implementation described by textbook.