Operating System

Discuss the basics of operating system functionality and structure, with examples of actual systems. The main contents are processes, synchronous communication between processes, scheduling, memory management, input and output, transaction processing, security, etc.

1 4 important features of the OS (04/08)

1. components of systems
   - management of memory, processes files, I/O systems.
   - OS provides the rule for the network based on the protocols
2. OS service
3. System call
   - it is the interface between OS and the user programs
4. System programs

2 Process and Kernel (04/15)

Today I learnt the process and kernel.

1. Process management
2. Virtual memory
3. What the fork function is
4. Process control brock
5. Thread models

3 Thread (04/22)

Today I learnt the thread. Thread is a part of the process and shares data and code. If two different threads access to the same memory, the conflict might happen. To avoid the conflict, codes or hardware are well-designed.

1. What is thread
   - Thread shares data and code but do not share stack
2. Three ways to manage some processes

- One is the way using shared memory
- The way using inter process communication
- Using signal

3. The way to avoid the conflict
   - Test and Set instruction
   - Swap instruction
   - Compare and Swap instruction
4. ABA problem
   - Avoud using tag bit
5. RCU

4 Deadlock (05/13)

Today's lecture is mainly about the deadlock, which is a problem in a computer program. It is hard to be notified, because it does not usualy occur the error. There are a lot of methods to prevent it. In this class, we learnt some of them.

1. Semaphores
   - The method to prevent a conflict
   - It has a method called counting semaphores, which counts the number of the waiting processes
   - Semaphore can be used to realize the synchronous processes(concurrent computing)
2. Deadlock
   - Deadlock is a state in which each member of a group is waiting for another member.
     - Two processes both need S and Q. One process takes S and the other takes Q. The former process waits until Q becomes free, and the latter process waits until S becomes free. However, Both need S and Q to finish processes, so both are dead.
   - Starvation is a state where a process is perpetually denied necessary resources to process its work
     - Deadlock is an example of the state where starvation occurs.
3. To prevent deadlock,
   - Critical regions
   - Monitors
   - These are the method which enables us safe programming, but the program is restricted to the safe shape.
4. graph
   - To show the probability of the deadlock in a programming code, a graph is used.

5 CPU Scheduling (05/20)

1. Basic Concept
   - CPU scheduling is a process which allows one process to use the CPU
   - main aim is to use CPU efficiently
   - Dispatcher and PCB
2. Five points to evaluate the scheduling
   - CPU utilization
     - cpu time = 100 % idle time
     - cpu time / system time
   - Wait time
     - The sum of time that processes wait on the queue until the processes finished
   - Through put
     - The amount of processes that CPU do in an unit time
   - Turn around time
     - the amount of time taken to complete a process or fulfill a request
   - Response time
     - The amount of time from when a request was submitted until the first response is produced.
3. Scheduling algorithm
   - FCFS
     - queue processes in the order that they arrived at the ready queue
   - SJF
     - select a process that has the shortest executing time
   - Priority scheduling
     - each process is assigned a priority
   - Round Robin

6 Real time process (05/27)

Today, I learnt some algorithms used in scheduling of process. Each algorithm is very simple and has both advantage and disadvantage.

1. Rate monotonic sheduling
2. Earliest deadline first scheduling
3. Global EDF and partition EDF

7 Memory management (06/03)

1. link
   - dynamic link and static link
     - dynamic link is a way that process gathers library before executed, and static link is a way that process while compiling.
2. How memory is used
   - user region, nmap region, and kernel region
   - overlay a way that uses memory effectively
   - Swapping
   - Fragmentation
     - compaction and other algorithms to prevent it
3. Virtual memory
   - Dynamic address translation
     - Virtual address corresponds to the pointer in language C
     - Pointer is translated into the fisical memory address using address translation table
     - In page table, there is an index that represents whether the column is valid or not

8 Virtual Memory 06/17

1. Translation lookaside buffer
   - the part of page table entry
2. Management of the virtual memory
   - Demand paging
     - Load to memory if it needs
   - Copy on write
     - Copy processes when write is executed
3. Page replacement algorithms
   - FIFO (first-in first-out)
   - LRU (least recently used)
4. Segmentation, working set model, and thrashing
   - Segments usually correspond to natural divisions of a program such as individual routines or data tables so segmentation
   - Working set is a concept which defines the amount of memory that a process requires in a given time interval.
   - thrashing occurs when computers' resources are overused which leads to constant paging fault

9 I/O 06/24

1. Virtual memory space in UNIX
   - sbrk systemcall
     - for example, malloc and free systemcall
   - mmap systemcall
     - map files to the virtual address space
2. I/O devices
   - Three kinds of interfaces
   - Character stream
     - input by 1 byte
     - mouse, syrial
   - block device
     - disk
   - network devidce
     - contact with ip and port number
   - RAID

10 File system 07/01

1. File
   - Files have thier permissions, which is stored in the parent directories.
   - Buffering makes treating files faster
   - Most OS today uses Acyclic-Graph directory
   - Access control list is the important for security.
2. Algorithms
   - allocate files using indexes
   - i Node

11 Protection, Batch system, and GPU 07/08

1. Protection
   - Domain structures in UNIX and Multics
   - Access Matrix
2. Batch systems
3. queing in batch systems
4. GPU