Principles of Operating Systems

Introduction

- An operating system is a program that controls and serves the execution of other programs.
- Functions of an OS:
- Process management;
- Memory management;
- Concurrency;
- Persistence.
- An OS should aim to provide a reliable and easy-to-use interface to applications over a diverse range of hardware backends in an efficient and protected manner.
- Layers of programs:
- Kernel layer;
- Service layer;
- Command layer and application layer.
- Modes of a CPU:
- Kernel mode;
- User mode.
- OS architectures:
- Monolithic;
- Layered;
- Microkernel;
- Modularized monolithic.

Processes

- A process is a running instance of a program.
- A single-thread process is characterized by:
- A memory space;

- A register context.
- Structure of a memory space:
- Text segment;
- Data segment;
- Heap;
- Stack.
- Life cycle of a process:
- New;
- Ready;
- Running;
- Blocked;
- Suspended ready;
- Suspended blocked;
- Terminated.
- Process management data structures:
- Process control block:
    - Process ID;
    - Execution state;
    - Register context;
    - Scheduling information;
    - Credentials;
    - Memory management information;
    - Accounting information;
    - Pointer to parent;
    - Pointers to children;
    - Pointers to resources.
- Process table, which maps a PID to a PCB;
- Process queues:
    - Ready queue;
    - Blocked queue;
    - Running pointer.

- Signals:
- A signal is an inter-process message with a predefined intention.
- Synchronicity of a signal:
  - Synchronous;
  - Asynchronous.
- Actions upon a signal:
  - Catching;
  - Ignoring;
  - Masking.
- A caught signal triggers the registered signal handler for its type.
- Actions upon SIGKILL and SIGSTOP cannot be overridden.

CPU Virtualization

- An OS virtualizes the CPU by context switches, serves processes via system calls, and preempts non-cooperative applications through interrupts.
- System calls:
- A system call is an applications request to the OS for a kernel service.
- Life cycle of a syscall:
  - Trap instruction;
  - Mode switch;
  - Invocation of syscall function;
  - Un-trap instruction;
  - Mode switch.
- Common syscalls:
  - Fork, that duplicates the current process under a new PID.
  - Exec, that loads a new program image into the current process;
  - Wait, that waits for the termination of child processes.
- Context switch:
- A context switch swaps the context of the current application process with that of another application process.

- Life cycle of a context switch:
  - Context saving;
  - State update of the current process;
  - Queue replacement of the current process;
  - Selection of the new process;
  - Context loading;
  - State update of the new process;
  - Mode switch.
- Overheads of a context switch:
  - Explicit overheads;
  - Implicit overheads:
  - Cache misses;
  - Memory misses.
- Interrupts:
- An interrupt is a hardware message delivered to the processor.
- Types of interrupts:
  - Hardware interrupts;
  - Synchronous self-interrupts;
  - Asynchronous self-interrupts;
  - Software interrupts.
- OS reclamation of the processor:
- Voluntarily via syscalls;
- Involuntarily via interrupts.

Scheduling

- Scheduling refers to the temporal allocation of processor resources to processes.
- Levels of scheduling:
- High level scheduling, which manages process creation;
- Middle level scheduling, which manages process suspension and resumption;

- Low level scheduling, which manages process scheduling onto and descheduling from processors.
- Related definitions of scheduling:
- Turnaround time, time between arrival and completion;
- Wait time, difference between turnaround time and service time;
- Response time, time between arrival and initial execution;
- Response ratio, ratio between projected turnaround time and service time.
- Basic scheduling algorithms:
- First in first out;
- Shortest job first;
- Highest response ratio first;
- Shortest time to completion first;
- Round robin.
- Multi-level feedback queues:
- There are multiple queues with descending priorities;
- Higher-priority jobs prevail;
- Jobs with the same priority are scheduled in a round-robin fashion;
- New jobs enter with the highest priority and get demoted if using up a quantum or total time quota at the level;
- The time quantum size grows exponentially as priority lowers;
- All jobs get boosted to the highest priority periodically.
- Fair share scheduling:
- Fair share scheduling aims to allocate a predefined fair share of CPU resources to each process.
- Fair share scheduling though lottery:
    - A fixed number of tickets are created;
    - Each process gets some tickets, indicating its fair share;
    - At each scheduling point, a lottery is performed, and the owner of the drawn ticket is scheduled.
- Completely fair scheduling by Linux:
    - The runtime of each process is recorded, normalized by its weight;

  - At each scheduling point, the process with the least normalized runtime is scheduled.

Concurrency

- Threading:
- A thread is a running sequence of instructions.
- A thread is characterized by a register context, a stack, and its membership in a process.
- A multi-thread process is characterized by a memory space and a set of member threads.
- Life cycle of a thread:
  - Creation;
  - Execution;
  - Termination;
  - Joining.
- Linux implements threads as light-weight processes, which are spawned by cloning instead of forking.
- Concurrency issues:
- A race condition is when the final value of a datum depends on the execution order of several processes that updates it concurrently.
- A critical section is a minimal section of code that updates a shared datum.
- An atomic operation is one that is guaranteed to finish without preemption once started.
- Locks:
- A lock is a flag indicating whether any thread is in a critical section for a datum.
- Properties of a lock:
  - It can be acquired by at most one thread at any time.
  - A thread cannot enter a critical section for a datum without acquiring the lock.
- Implementations of a lock:
  - Interrupt masking:

  - Interrupts are marked during the atomic;
  - This does not work on multi-core systems and is vulnerable to non-cooperation.
  - Atomic instructions:
  - The test-and-set instruction returns the current value of a variable and updates it with a new one atomically.
  - The compare-and-swap instruction tests if the variable is equal to a given expectation, updates it with a new value if equal, and returns the original value atomically.
  - Mutexes:
  - The mutex is the POSIX implementation of the lock.
  - The OS blocks a thread attempting to acquire a locked lock.
- Synchronization:
- Synchronization is the guarantee of execution order between or among a set of statements across threads.
- Condition variables:
  - A conditional variable sets up a pool for resource distribution;
  - A wait adds a process to the pool and blocks it;
  - A signal pops a process from the pool and unblocks it.
- Semaphores:
  - A semaphore sets up a queue for resource distribution and an integer flag;
  - A wait decrements the flag, and adds a process to the queue and blocks it if the flag is now negative;
  - A post increments that flag, pops the head of the queue, and unblocks it.
- Deadlocks:
- Necessary conditions for a deadlock:
  - Mutual exclusion;
  - Hold-and-wait;
  - No preemption;
  - Circular wait.
- Dealing with deadlocks:

  - Prevention:
  - Atomic request bags;
  - Preemption;
  - Resource ordering;
  - Avoidance:
  - Avoidance by scheduling;
  - Banker's algorithm, i.e. avoidance by granting;
  - Detection and recovery:
  - Abort-all;
  - Abort-till-success.

Memory Virtualization

- Purposes of memory virtualization:
- Memory sharing;
- Accomodation of oversized memory spaces;
- Memory integrity.
- Each application sees a contiguous memory space starting from 0 through its virtual addresses.
- Memory address translation:
- Memory management unit:
  - Translates virtual addresses to physical addresses;
  - Checks for segmentation faults.
- OS:
  - Manages address space profiles;
  - Maintains memory allocation records.
- Segmentation:
- Segmentation divides an address space to fixed-length intervals called segments and allocates them memory independently and contiguously per their used sized.
- The first few bits of a logical address represent the segment id, and the remainder stores the offset.

- Each process has a segment table where an entry describes each of its segments.
- Organization of a segment table entry:
  - Base;
  - Bound;
  - Flags:
  - Grow bit;
  - Right bits:
    - Read bit;
    - Write bit;
    - Execute bit;
  - Valid bit.
- Fragmentation under segmentation:
  - Memory fragmentation is the phenonmenon that some free space cannot be used.
  - Combating fragmentation:
  - Coalescing;
  - Compaction;
  - Memory allocation algorithms:
    - Best-fit;
    - Worst-fit;
    - First-fit;
    - Next-fit.
- Paging:
- Paging divides an address space into fixed-length intervals called pages and allocates them memory independently and contiguously in fixed frames of the same length.
- The first few bits of a logcial address represent the page id, while the remainder stores the offset.
- Each process has a page table where an entry describes each of its pages.
- Organization of a page table entry:
  - Frame id;

  - Flags:
  - Right bits;
  - Valid bit;
  - Present bit;
  - Use bit.
- Under paging, each present page is allocated to a frame.
- Practical paging:
- Under naive paging, each logical memory access incurs two physical accesses, and the page table is exceedingly large in a modern architecture.
- Elements of practical paging:
  - Translation lookaside buffer;
  - Hierarchical page tables.
- Translation lookaside buffer (TLB):
  - TLB is a dedicated cache for page tables in the MMU;
  - TLB is associative;
  - TLB supports hardware parallel search;
  - A typical size of a TLB is 128 entries.
- Page faults and replacement:
- A page fault is a page access failure caused by unpresence.
- Solving a page fault:
  - OS requests to load the page from disk and blocks the fault-triggering process;
  - The page is loaded, and an interrupt triggers OS to ready the process;
  - The process resumes, re-attempts the access, causes a TLB update, and gets the data.
- Page replacement policies:
  - First-in-first-out;
  - Least-frequently-used;
  - Least-recently-used;
  - Clock replacement.
- Thrashing is the phenonmenon when over-subscription causes frequent out-swapping of pages of processes waiting for page loads.

Persistence

- A computer typically uses external persistence devices, like hard drives, solid-state drives, and optical disks.
- A persistence device typically provides an interface through which data is organized in an array of fixed-sized intervals called logical blocks.
- A file system is a component of an OS that provides an interface over persistence devices through which data is organized in variable-lengthed entities indexable by a logical path, called files.
- Functions of a file system:
- File management;
- Space management;
- File integrity;
- File security.
- File:
- A file is an abstract data type for persistence.
- Attributes of a file:
  - Path;
  - Id;
  - Location;
  - Size;
  - Accessibility;
  - Date, time, and ownership;
  - Reference count.
- Operations on a file:
  - Create;
  - Delete;
  - Open;
  - Close;
  - Read;
  - Write.
- The name of a file is partially stored in its directory tree, all other attributes are in its file control block (FCB).

- A directory is a file storing a set of mappings from basenames to file ids.
- Links:
- Hard links:
  - A hard link to a file is a directory entry with a different path but the same id.
  - Directories cannot have hard links to avoid non-terminating traversals.
  - Hard links are restricted to the same partition by their use of file ids.
- A soft link to a file is another file storing the path to the linked file.
- File system organization:
- Superblock;
- FCB table;
- FCB bitmap;
- Block bitmap.
- Superblock:
- The superblock of a file system stores the file system id, the number of FCBs, and pointers to the FCB table, the FCB bitmap, and the block bitmap.
- The superblock is usually the first block of a file system.
- The FCB table is a fixed-length array of FCBs.
- The FCB bitmap is a bitmap where each bit indicates the validity of an FCB in the FCB table.
- The block bitmap is a bitmap where each bit indicates the availability of a logical block.
- Block management of a file:
- Linked list of blocks;
- Block allocation table, i.e. a centralized linked list;
- Multi-level index, the preferred approach.
- Free space management:
- Linked free list;
- Centralized free list;
- Block bitmap, the preferred approach.
- File security:

- Upon an open, the file system check accessbilities, creates an entry in the open file table, and records the granted rights.
- Upon a read, the OS checks the access rights.
- File integrity:
- An OS typically provides utilities for file system backup and consistency checking.
- A journaling file system performs each metadata update by logging, write-out, and completing to ensure consistency.