**Aidan Levyne, Jacob Waxman, Kyle Strougo, Olivia Nye**

**Objectives**

## **1.** **Interesting terms and concepts (aka new fancy words you learned or revisited in this course)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Term and/or brief summary** | **Computer system examples** | **Examples from different academic areas (including other branches of CS)** | **Examples from everyday life** |
| Caching (storing a copy of recently accessed data in a storage that’s faster than the primary storage). | · Processor cache maintains small subset of data from main memory.    · Flash drive is used as cache for HDD. | * When a device enables wifi, recently used networks are shown first | * iPhones store information and contact info for recent phone calls in a stack on the “Recents” page of the “Phone” app. |
| Virtualizing the CPU (OS provides the illusion that there are an infinite number of processors, every program can have its own CPU) | * With CPU virtualization, the system utilizes privilege levels to protect restricted data/operations   + This is done utilizing the trap manager | * When a device uses a network, there is an illusion that you are the only client on that network | * With VR, users are given the illusion that there are an infinite number of realities |
| Virtualizing the memory (the illusion that every process gets its own private address space | * Through address translation, the system can utilize segmentation and page tables to direct the system to the location of the data | * A desktop may contain files, however this is just the illusion of the location of these files in memory * The true location of these files is much more complicated | * Similar to the utilization of page tables to direct the system to a location in memory, a person can navigate an airport using information screens to locate their terminal/gate |
| Concurrency (Utilized by multi-threaded programs to execute multiple threads within a single program in a particular order and in the same address space) | * A program with multiple threads will commonly utilize a critical section with universal resources/data, and locks to execute in an efficient/systematic matter * This is displayed in the dining philosophers problem | * A network will commonly have multiple computers/ clients   + With this we can expect “Grid Computing” | * An elevator system in a building * Multiple people at different levels are requesting access to a single/shared resource |
| Process State (Program is given a status where it can be either running, blocked, or ready. Used to optimize CPU availability) | * Any multithreaded/ multitasking computer can utilize this to manage many processes at once | * In networking, when establishing connection many times a client waits for the state of a host to be available | * In airports many planes share a single runway * In this case, planes are given a specific state when ready for takeoff/ landing |
|  |  |  |  |

## **2.** **Useful problem-solving approaches**

|  |  |  |  |
| --- | --- | --- | --- |
| **Brief statement** | **Computer system examples** | **Examples from different academic areas (including other branches of CS)** | **Examples from everyday life** |
| Periodically check the system status in order to detect errors early. | Error detection is one of the many services that the OS can provide. When the OS regains control from other processes, it should check for errors before yielding the CPU to any other processes in order to ensure early detection of errors created by preceding processes. | Periodically check the status of a biological experiment to catch obvious problems early. | During the current global health crisis, it is good practice to periodically check in with friends and family that you may have been exposed to and check to see if they are showing any symptoms of covid-19 or have tested positive. |
| Round robin (each process gets to run for a set amount of time, then the next process gets to do the same and so on… going through the queue of processes repeatedly until all processes are finished) | Round Robin CPU scheduling for processes of equal priority in a multi-level feedback queue. | Round robin scheduling is one of the scheduling policies that can be used on a network level to help routers choose the next packet to send to a link. | When sharing a bottle of water between multiple glasses, you pour the same small amount into each glass, repeating until all glasses have been removed because they are full or until you run out of resources (water). |
| Segmentation (dividing a resource into segments of a variable size. Allows each segment to be considered/controlled separately in some respects while also allowing them to share certain resources) | Segmentation in virtualized memory management allows memory to be split into a number of segments that each can be located independently. We can control access permissions of each segment separately and they each can have their own heap and stack, but the segments can share code blocks. This is especially useful for shared libraries and forked processes. | Segmentation is used in image processing in computer vision to segment digital images into sets of pixels, which can make the image easier to analyze and/or change. | Segmentation could be likened to dividing the employees at a company or in a department into teams (segments). Each team may have different permissions and may be located, considered, and controlled separately, but all teams can share certain resources. |
| First in first out eviction policy | TLB eviction policy which evicts the first thread added once space is needed . | FIFO is a term used in the accounting world, when estimating the cost of goods sold, you can apply the FIFO strategy and assume you sold the goods which have been stored in inventory for the longest time. | Lines at a store. First in line pays first, last in line pays last. |
| Least recently used used eviction policy | TLB eviction policy which evicts the thread least recently used | A cache of internet browsing history. Most recently visited cites on top, least recently used at the end of the list | Organizing your closet by LRU strategy, the clothes you just wore would be in the front, clothes you haven't worn in a while will be in the back |
| Optimal eviction policy | TLB eviction policy which evicts the thread which won't be used for the longest period of time | In business analytics, an excel function uses the Optimal strategy to organize job scheduling and optimize time efficiency by doing the jobs that needed to be done ASAP first and then those that wont need to be done for a while | Time scheduling. It is the most efficient to do you school work that needs to be done tomorrow and then the work not needed to be done for a long time |

## **3.** **Neat solutions, hacks, design tricks**

|  |  |  |  |
| --- | --- | --- | --- |
| **Brief statement** | **Computer system examples** | **Examples from different academic areas (including other branches of CS)** | **Examples from everyday life** |
| Devising separate solutions for initial system startup, system resume, and normal system operation. | Different types of actions are performed at OS booting, standard OS operation, and OS shutdown. |  | Similarly, different types of actions are performed at starting a car, continuous running of a car, and car shutoff. |
| Ticket Lock | Using fetch-and-add, or a ‘turn’ queue, threads use atomic operations to decide who has access to critical data. | Based on a system’s stack and heap, processes can only successfully change specific data if given priority. | In a line at the Deli or DMV, people wait their turn to access important resources (in these cases, the attention of service) because only a limited number of people can execute commands at once. |
| Context Switch | Context switches occur when the CPU ensures data including registers, the stack pointer, and the program counter is saved, though a different thread is allowed to execute. | In other fields of study, specific projects or commands often need to be put on hold to address others. Information is saved such that, after other processes allow it, these halted projects can resume as they were left off. | If for example, you are reading an article on your phone and suddenly you receive an important text, you can leave the article, reply to the text and resume where you left off. |
| Multilevel feedback queue | The multilevel feedback queue is an approach to scheduling that attempts to minimize turnaround time, or the total time allotted from when a process starts to when it terminates, while at the same time minimize response time, or the time it takes for a process to begin after the user initiates activity. Processes move between queue levels based on time, activity and requests. | In medical or chemical research, new information often reveals changes in perceived progress and response time of tasks. This information changes the priority of certain research processes above others in an attempt to be more effective and responsive towards reaching goals. | If a certain activity is unresponsive or boring after a long period of time, we naturally switch to something else and come back to it in order to be most productive and engaged. |
| Yield | Yield is a command that moves the currently running thread to the end of the active queue under some condition. | If some scientific project is executed until halted by an unmet requirement or condition, the process isn’t terminated, but resumed later, after processes necessary to meet this condition have executed. | Similarly, if you are trying to complete a homework assignment and a specific question asks you for a reaction to another assignment you haven’t completed yet, you will wait to work on that assignment and retry the question after. |