CSc 139 HW01

Alec Resha

1. What is essential about an operating system?
    1. An OS has to manage memory and multiple running processes. Very helpful to offer a GUI for a computer as well
2. Why is it meaningful in a computing environment to use an OS? What key added value does an OS provide?
    1. Operating system manages multiple programs and processes and running at once. Without an OS, a computer could only run one program at a time, and could only run them sequentially. Errors will have more widespread effects.
3. What is a possible constraint, value, and use of a computer without an OS?
    1. Very little value, cannot multitask and can only do one thing at a time.
    2. Primary use of a computer with no OS would be doing a lot of low level tasks with very little computing power (arduino, etc)
4. Explaing the essence of a true interrupt. What causes interrupts? How can an application know that it has been interrupted?
5. List at least 5 key management functions of an OS.
6. Contrast info in main memory with secondary storage.
    1. Main Memory is directly accessible by the CPU, accessible quickly but expensive
    2. Secondary memory is much easier to store large amounts of data, slower than main memory and not directly accessible by CPU (has to be loaded into mainmemory)
7. Acronym EULA? Why use EULAs? Discuss typical programmers cost of EULA for Unix and Linux.
    1. End User License Agreement.
    2. Most common use is to remove the developer from liability.
    3. EULA for UNIX and Linux are free.
8. Characterize an OS's function of Resource allocation, sharing, accounting, protection, and Security
9. Describe System Call. How can they be made to fit specific needs? List a sample system call.
    1. Way that a program interfaces with the OS.
    2. Can fit a specific need based on inputs to the call.
    3. example: Read()
10. Explain in which programming language an OS could be impletemented. Argue Why.
    1. C is the most common language for coding an OS.
    2. It was specifically made in order to write Operating Systems in a higher level language an assembly language. Extremely lightweight to run, and is already the standard language for OS's.
11. Outline hierarchy of at least 5 storage resources and technologies for holding info. Mention or sort by speed.
    1. Ram - Nearly instant access, expensive for more, temporary storage

    2. SSD - Very fast, degrades slightly sooner than HDD memory but far faster
    3. HDD - Relatively fast, degrades slowly
    4. Optical Disk - Very slow by modern standard, physical disk read via laser
    5. Tape Drive - Extremely slow but can store data for very long periods of time
12. Informal Definition of OS kernal.
    1. The kernal is always in memory, it is a program that controls all other programs and processes
13. Describe what is essential in a Real Time compute environment
    1. Able to process a large amount of data, fast response time
14. What essential steps to get an OS to work when a computer is turned on
    1. Make sure the kernal is stored in a fixed location that the hardwere BIOS knows where to look, kernal gets loaded into memory, and kernal has to have the ability to start everything from there.
15. Characterize in detail what a process is. Describe: typical process needs, types of processes, causes of process start and end.
    1. A process is the part of a program that is being run currently.
    2. 4 types are input, processing, output, and Storage
    3. Processes are usually started by the kernal loading the program into Memory with a system call
    4. Normally terminated when the process is finished running with the system call of exit(), OS can also throw the exit() call to stop a process
16. What is an Idle Loop? Why is it part of an OS? Should there still be an idle loop?
    1. When there is no user input or work to do, the OS idles for a moment before checking for more work to do.
    2. If it wasn't part of an OS, the OS would shut down when there was no input.
    3. Yes there should still be an idle loop
17. What is a utility program? Why have one? Give an example. Are they integral part of an OS?
    1. Utility programs are used to resolve common issues that occur. They allow the user to fix a problem that the OS doesn't throw as an error.
    2. Disk Defragmentation is a utility program
    3. Not an integral part since an OS can run without them, but they are very helpful
18. Explain Disk Fragmentation. What causes it? What is the impact on user programs?
    1. When storing data, all the data for a program can get spread around sections of memory.
    2. The initial area has it's own section, but it is likely that the space after it will be allocated to something else so more memory is spread out.
    3. Slows down IO operations for the relevant programs since it has to look up remaining storage location.
19. What is DMA? Why does a computer system support DMA? What is its main advantage?
    1. DMA is Direct Memory Access. It allows a program to access data directly from main memory.
    2. Allowed for faster data access.
    3. Main advantage is it is faster than going through the CPU

20. List some similarities and differences between the Unix and Linux operating systems.
    1. All Linux and Unix operating systems have access to the Linux source code
    2. Linux is entirely open source, some UNIX systems are proprietary or have Licenses
    3. Linux managed by users, UNIX often managed by organizations or businesses
    4. Both written in C