Samantha Pope - CS6013 - A2: Reading Vocabulary Assignment

I defined all of the following terms using my own words. On words I was particularly confused about, I added the google definition so that I can use this to study for the midterm in the future. Please disregard these "google definitions" that have red text.

Hz -

Hz is a frequency unit. In the CPU, it can measure how many cycles per second can occur. In class we discussed that the CPU can run in GigaHz, which means it is running about 3 billion instructions per second.

Kilo-

Kilo means thousand 10^3

Mega-

Mega means million 10^6

Giga-

Giga means billion 10^9

Tera-

Tera means trillion 10^{12}

elf (in terms of Unix programs)-

Elf: executable and linkable format. It is a standard format within unix programs which allows the operating system to know exactly where to look in the file to find what it needs to run. Elf can include standardization for executables, object files, shared libraries, etc. It helps everything communicate with each other.

From google: "In essence, an ELF file is a container for the code and data that make up programs and libraries, structured in a way that the operating system can efficiently use it to perform tasks, run applications, or recover from errors."

Thrashing-

Thrashing would occur if we are constantly doing context switching, and never getting any code done within the process. Context switching costs cycles, it is running a code to switch between them. You could just run this code forever and stay in relatively the same spot in all of your programs you are "running". This can happen when you have insufficient physical memory available.

Virtual Memory-

Virtual memory is what we do so that each program thinks it starts at 0x00 and has a continuous stream of memory. This allows the user to say things like "hey move this

instruction 20 spots down from 0". In reality, our code starts at an ugly address in real memory (like 0x 5FAB228) that the operating system hides from us. This lets us think we are accessing memory, but we are not manually

Virtual Address Space-

Address space is a range of memory that the process can access. The virtual address space is the range of virtual addresses that a program can use. The OS translates these virtual addresses to real RAM memory addresses.

Virtual Address-

The virtual address is the "fake" address that the user can manipulate in the program. It does not mean the actual address in RAM, which would look a lot uglier. This allows the OS to isolate memory space of different programs and allocate it wherever there are empty spots (even if it is out of order)

Virtual Machine-

A virtual machine is like what we used in the chat client, it looked like we had an android phone on our computer but it wasn't real. This allowed us to test the functionality of our code and how it would run in an android app.

Time Sharing-

Time sharing is when the CPU "multitasks". The CPU is actually only running one thing at a time, however it switches between tasks so quickly that to us, it looks like multiple programs are running at the same time when in actuality they are not.

Digital (Discrete) vs Analog-

Digital signals in computing are signals that are in binary and can be read by the computer system. Analog signals are continuous data and vary in signals. For example, analog TV signals.

Kernel vs OS-

The kernel is the core part of an operating system, it manages hardware resources and provides basic services for the OS. The operating system includes the kernel and other software and utilities to manage computer hardware and gives services for apps.

Von Neumann-

Famous guy in computing - know how to pronounce his name. This is a type of architecture - it outlines a design fo computers. It includes a central processing/control unit that stores data and instructions/manages memory. It also has basic arithmetic functionality built in. This type of structure is still how we model computer organization today.

Integrated Circuit (IC)-

Set of circuits on a chip that can conduct electricity. They are the building blocks used in most modern electronic devices.

State Machine (FSA)-

FSA: Finite state automaton. This just says that you can only be in one state at one time. For example a process can only be ready, zombie, running, or an embryo at one time. It cannot both be a zombie and running.

Hardware (or program) privilege level-

This is the ring model we discussed in class. The closer you get to the center, the more privileged a layer is. A program would operate on the most outer layer, where it cannot directly access things. If it needed to access certain things, then you would ask the operating system or things that operate on the inner layers to get access to these functions/processes.

Context Switch-

Context switching is what lets us have multiple things that access the CPU resources at one time. This can happen when we write a syscall or something that needs to bother the OS, the OS can save its place and then complete our syscall and go back.

Lazy loading (of program, of memory, etc)-

Lazy loading means that we wait to initialize a resource until the user actually needs it. This allows us to more efficiently use resources and not load in things that we will never use. Kind of a deal with it when it's asked of us.

PID - Process ID-

This is a unique ID that is assigned by the OS to each process "running" on the computer. You can ask the PID of something to know what process is running on certain ports, or know if we are talking to a child/parent process when using fork() and exec().

DMA - direct memory access-

Direct memory access allows certain systems to access main system memory independently of the CPU. This is dangerous, but it can also make things run faster since they don't have to ask the CPU every single time it needs to change something. This is a privilege given to storage devices, graphics card, sound card, webcams, etc.

MMU / TLB-

MMSU (Memory Management Unit) is a component of the CPU that handles virtual to physical address translations. It maps the virtual memory to actual memory addresses.

TLB(Translation Lookaside Buffer) is a cache within the MMu that is designed to speed up the address translation process. It stores recent versions of virtual addresses to quickly provide mappings and reduce access times. Think of it like your cache in your DNS resolver, it stores things we ask for a lot so it can immediately give us the right address.

Daemon-

Daemon is always running in the background and performing specific operations at predefined times or when certain things happen. (think of literal demon that performs tasks in the background always and then can respond when the devil needs them).

Symbol (with respect to Code, object files, assembly)-

Symbols, in this context, are names that represent an entity in code. In object files and assembly files, symbols map to a specific memory address and can be placeholders for addresses that the linker can figure out and decode.

Shared Library-

Shared library is collection of functions and resources that can be shared and used by programs. Think of stlib = standard library that is shared among programs and lets you access data structures/object types.