Matt Kennedy
CIS 452
Homework 1

1. **What is the purpose of interrupts?**
   The purpose of an interrupt is to interrupt whatever the cpu is doing currently so it can work on multiple things without forcing slowness or waiting times to the user.

2. **What is the main difference between a trap and an interrupt?**
   From the book, “Atrap (or an exception) is a software-generated interrupt caused either by an error (for example, division by zero or invalid memory access) or by a specific request from a user program that an operating-system service be performed.” From further reading on Stack Overflow, this is also a way to do system calls because these run with higher priority than interrupts.
   This differs from interrupts, which are defined in the book as “the signalling of an event”, which seems like a broader term than trap. Another difference is traps are exceptions, whereas interrupts are more general and encapsulating for interrupts in the cpu.

3. **Traps can be generated intentionally by a user program. For what purpose?**
   As stated in the last question, traps are rated at a higher priority than interrupts, and can be used to perform a system call. This also swaps the mode to kernel mode and sets the trap bit to 0.

4. **What is the purpose of the command interpreter?**
   The command interpreter allows an interface for the user to type commands for the operating system to execute.

5. **Why is it usually separate from the kernel?**
   From research and from class, there are two main reasons - First, the command interpreter may be subject to change, which can cause issues if it is part of the kernel. Second, the kernel provides all the essentials of an operating system as it’s core, and will always stay in memory or “active”. Because of this, the kernel should be as small as possible. Command interpreter’s aren’t necessary to stay in memory constantly, so they are included in the OS, just not in the kernel.

6. **What is contained in the /arch subdirectory of the linux source code?**
   After looking at Linus Torvald’s GitHub and viewing the linux repository (which has over 800,000 commits wowee wowzer), it appears to be different instruction sets which all have their own kernel and (this is a total guess because this is week 1 of the course) entire operating system based around that instruction set.

7. **How are these contents used?**

Compatibility - if a program is compiled in one architecture, the set of instructions may not run as well, or at all, on a different architecture. The point of having these kernels available in multiple architectures is so that programs can be run on different operating systems, and not everyone needs to have the same architecture.