CSC 452 Review 3

https://github.com/Jon-Davis/CSC452

Threads and Processes

Threads

```
void * myThread(void *vargp){
        // Do something
}

int main(){
        pthread_t thread_id;
        pthread_create(&thread_id, NULL, myThread, NULL);
        pthread_join(thread_id, NULL);
}
```

Processes

```
Int main() {
        pid_t  pid;
        fork();
        pid = getpid();
        If (pid == 0){
                // I am the newly created process
        } else {
                // I am the parent process
        }
}
```

Communication

Threads

Threads share a common address space.

If one thread creates a variable, all other threads are capable of seeing it.

Processes

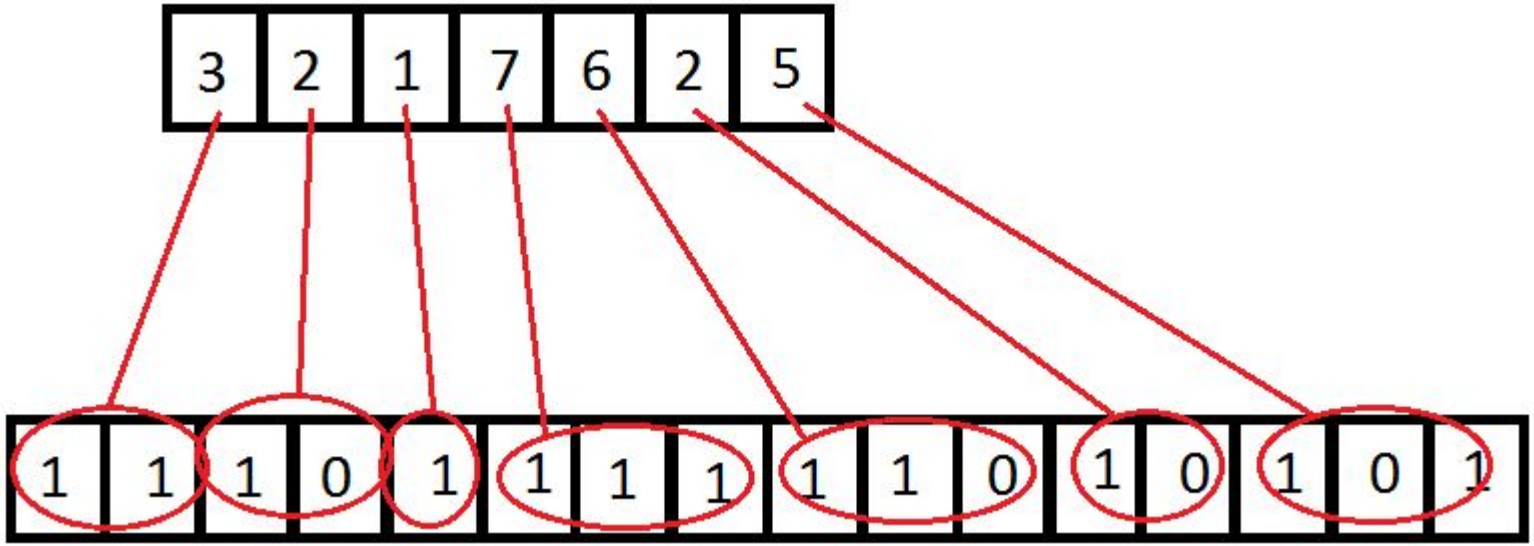
Processes have their own address spaces.

In order to share memory, processes use memory maps. This can be achieved by using functions like mmap.

Integers as Arrays

If we have a short with the value 1110,1111,1101,0101, we could interpret it as 61,397

Or we can interpret it as a struct or array of lesser values.



Static

A Static global variable can only be accessed within the file it is declared. Similar to the private keyword found in other languages.

A Static variable declared in a function saves its state between calls.

```
static int private_int = 0;
void iterate() {
        static int value = 0;
        printf("%d\n", value++);
}
```

```
Int main() {
        iterate(); // prints 0
        iterate(); // prints 1
        iterate(); // prints 2
}
```