Threads and Synchronization

The Pthread Library

- `int pthread_equal (pthread_t tid1, pthread_t tid2);`
- `pthread_t pthread_self (void);`
- `int pthread_create (pthread_t *tid, const pthread_attr_t *attr, void *(*startRoutine)(void *), void *arg);`
- `void pthread_exit (void *valuePtr);`
- `int pthread_join (pthread_t tid, void **valuePtr);`
- `int pthread_cancel (pthread_t tid);`
- `int pthread_detach (pthread_t tid);`

Semaphore POSIX

- `int sem_init (sem_t *sem, int pshared, unsigned int value);`
- `int sem_wait (sem_t *sem);`
- `int sem_trywait (sem_t *sem);`
- `int sem_post (sem_t *sem);`
- `int sem_getvalue (sem_t *sem, int *valP);`
- `int sem_destroy (sem_t *sem);`

Mutex POSIX

- `int pthread_mutex_init (pthread_mutex_t *mutex, const pthread_mutexattr_t *attr);`
- `int pthread_mutex_lock (pthread_mutex_t *mutex);`
- `int pthread_mutex_trylock (pthread_mutex_t *mutex);`
- `int pthread_mutex_unlock (pthread_mutex_t *mutex);`
- `int pthread_mutex_destroy (pthread_mutex_t *mutex);`