

Taller #1: Threads

1. Existen muchos paquetes que nos permiten hacer esto en C, como por ejemplo POSIX Threads, CLthreads, LinuxThreads.
2. Mutex nos permite que puede usarse por diferentes threads de un programa un mismo recurso, y le impide el uso a un hilo si otro thread ya lo esta utilizando.
3. Al mismo tiempo no pueden usar el mismo recurso, ya que si estos se encuentran sobre este recurso compartido los hilos que no estan usandolo permanecen bloqueados *lock* hasta que el thread libere este recurso.

Ejercicios

1. Podemos ver en la siguiente imagen del funcionamiento de los threads, vemos los 6 elementos de los array, ademas del array que contiene el XOR por elemento.

```
ema@ubuntu:~/Documents/Arqui2/Taller1$ ./pregunta1
Creado.
Creado.
Creado.
Arreglo 1 = [ 103 105 115 255 41 186 ]
Arreglo 2 = [ 198 81 74 236 205 171 ]
XOR = [ 161 56 57 19 41 186 ]
```

2. Se puede ver aca que los hilos van alternándose para acceder a los recursos o funciones creadas.

```
ema@ubuntu:~/Documents/Arqui2/Taller1$ ./pregunta2
Creado.
Funcion 1
Variable: 1
Saliendo de la funcion 1

Creado.
Funcion 2
Variable: 2
Saliendo fde la funcion 2
```

Bibliografia

Procesos e hilos en C de Unix/Linux. (s. f.). ChuiDiag. Recuperado 3 de marzo de 2021

UC3M. (s. f.). 11.2. Hilos. iT. Recuperado 3 de marzo de 2021