

## Semaphores

- Semaphore is a variable whose value is one , its value decreases on acquire function to zero(lock) , and increases when release
- Semaphores are same as lock but one difference is that in semaphore 2 threads are allowed inside function at a time
- Thus, semaphores allow multiple threads
- If semaphore value is one then one thread will enter the function , if two then 2 values will enter
- If its value is zero no thread is allowed inside function , if the value is non-zero then that no.of threads are allowed inside the function
- If they are many threads then they'll wait in a queue
- Below is the program that allows 2 threads to enter a function at the same time

```
from threading import *
from time import *

def display(str1):
    l.acquire()
    for x in str1:
        print(x)
        sleep(1)
    l.release()

l = Semaphore(2)

t1 = Thread(target=display, args=('HELLO WORLD',))
t2 = Thread(target=display, args=('you are welcome',))
t3 = Thread(target=display, args=('0123456789',))

t1.start()
t2.start()
t3.start()

t1.join()
t2.join()
t3.join()
```