1. How many semaphores are available in DLXOS?

Answer: 32

2. How many locks are available in DLXOS?

Answer: 64

3. How is a semaphore created in DLXOS?

Answer: SemCreate() calls SemInit() which

starts the process queue and sets the counter.

Once it is initialized, SemInit() passes back a

handle to the semaphore to SemCreate().

4. Explain how sem\_signal and sem\_wait work.

Answer:

sem\_signal: calls SemHandleSignal, which calls SemSignal,

which increments the semaphore and wakes a sleeping process

if any are waiting (think of walmart shelf being added to)

sem\_wait: calls SemHandleWait, which calls SemWait, which

decreases the semaphore value and if it goes below zero,

it will sleep and wait for one to be signaled (think of

something being taken from walmart shelf, and if they are

empty, the consumer goes to sleep and waits for something

to be added to the shelves)

5. What is a handle and how is it used to access semaphores in DLXOS

Answer: Handles are a pointer to a data structure called "Sem" that stores the value

of the semaphore and another value indicating whether it is in use

or not called "inuse". When a process is using the semaphore, it will

change "inuse" to signal that the semahpore is "locked" so that if

another process tries to access it, then that new process

will see that it is in use. The actual numerical value of the sempahore is

maintained with the "count" field of the data structure. (think of handle

as a global variable so that all processes can access the same semaphore)