

unit_count	3				
producer_count	2				
consumer_ount	2				
can_accesss_queue					
can_access_next_unit					
can_consume	0				
next_unit	3	1	2	3	
productor 0	productro 1	consumidor 0	consumidor 1	Hilo Principal	
while true		
declare my_unit = 0	while true		
lock(can_access_queue)	declare my_unit = 0	while true	...		
next_unit < unit_count	... wait (lock de p0)	wait(can_consume)	while true	join_threads	
next_unit ++	... wait (lock de p0)	.. wait	wait(can_consume)	.. wait	
my_unit = next_unit	... wait (lock de p0)	.. wait	.. wait	.. wait	
unlock(can_access_new_unit)	... wait (lock de p0)	.. wait	.. wait	.. wait	
lock(can_access_queue)	lock(can_access_queue)	.. wait	.. wait	.. wait	
enqueue(queue, my_unit)	next_unit < unit_count	.. wait	.. wait	.. wait	
unlock(can_access_queue)	next_unit ++	.. wait	.. wait	.. wait	
print()	my_unit = next_unit	.. wait	.. wait	.. wait	
signal(can_consume)	unlock(can_access_new_unit)	.. wait	.. wait	.. wait	
while true	lock(can_access_queue)	.. wait	.. wait	.. wait	
declare my_unit = 0	enqueue(queue, my_unit)	.. wait	.. wait	.. wait	
lock(can_access_queue)	unlock(can_access_queue)	.. wait	.. wait	.. wait	
next_unit < unit_count	print()	lock(can_access_queue)	.. wait	.. wait	
next_unit ++	signal(can_consume)	declare my_unit = dequeue	.. wait	.. wait	
my_unit = next_unit	while true	unlock(can_access_queue)	.. wait	.. wait	
unlock(can_access_new_unit)	declare my_unit = 0	my_unit == -1	lock(can_access_queue)	.. wait	
... wait	unlock(can_access_new_unit)	print	my_unit == -1	.. wait	

... wait	break	while true	print	.. wait	
lock(can_access_queue)	... end	wait(can_consume)	while true	join(producers)	
enqueue(queue, my_unit)	... end	.. wait	wait(can_consume)	lock(can_access_queue)	
unlock(can_access	... end	.. wait	.. wait	enqueue(queue,-1)	
print()	... end	.. wait	.. wait	unlock(can_access_queue)	
signal(can_consume)	... end	.. wait	.. wait	signal(can_consume)	
while true	... end	lock(can_access_queue)	.. wait	.. wait	
declare my_unit = 0	... end	declare my_unit = dequeue	.. wait	.. wait	
lock(can_access_queue)	... end	unlock(can_access_queue)	.. wait	.. wait	
next_unit < unit_count	... end	my_unit == -1	.. wait	.. wait	
unlock(can_access_new_unit)	... end	print()	.. wait	.. wait	
break	... end	while true	.. wait	.. wait	
... end	... end	wait(can_consume)	.. wait	.. wait	
		.. wait	lock(can_access_queue)	.. wait	
		.. wait	declare my_unit = dequeue	.. wait	
		.. wait	unlock(can_access_queue)	.. wait	
		.. wait	my_unit == -1	lock(can_access_queue)	
		.. wait	break	enqueue(queue,-1)	
		.. wait	end	unlock(can_access_queue)	
		.. wait		signal(can_consume)	
		lock(can_access_queue)		.. wait	
		declare my_unit = dequeue		.. wait	
		unlock(can_access_queue)		.. wait	
		my_unit == -1		.. wait	
		break		.. wait	
		end		join_threads(consumers)	
				end	