



CONCURRENT QUEUE						
Remove fro	m queue					
1 2	<pre>typedef structnode_t {     int value;</pre>					
3 4 5	<pre>structnode_t *next; ) node_t;</pre>					
67	typedef structqueue_t { node_t *head; node_t trail.					
9 10	pthread_mutex_t headLock; pthread_mutex_t tailLock;					
11 12	) queue_t;					
13 14 15	<pre>void gueue_init(queue_t *q) {     node_t *tmp = malloc(sizeof(node_t));     tmp-&gt;next = NUL;</pre>					
16 17 18	<pre>q-&gt;head = q-&gt;tail = tmp; pthread mutex init(sq-&gt;headLock, NULL); pthread mutex init(sq-&gt;tailLock, NULL);</pre>					
19 20 (Cont.)	)					
February 13, 2019	TCSS422: Operating Systems [Winter 2019] School of Engineering and Technology. I Jouerity of Washington - Taroma	L10.5				









































EXECUTION TRACE:									
	<i>T</i> <sub>c1</sub>	State	T <sub>c2</sub>	State		State	Count	Comment	
	c1	Running		Ready		Ready	0		
	c2	Running		Ready		Ready	0		
	c3	Sleep		Ready		Ready	0	Nothing to get	
Legend		Sleep	c1	Running		Ready	0		
c1/p1-lock		Sleep	c2	Running		Ready	0		
c2/p2- check var		Sleep	c3	Sleep		Ready	0	Nothing to get	
c3/p3- wait		Sleep		Sleep	p1	Running	0		
c4- put()		Sleep		Sleep	p2	Running	0		
n4 got()	_	Sleep		Sleep	p4	Running	1	Buffer now full	
p4-gei()	5	Ready		Sleep	p5	Running	1	T <sub>c1</sub> awoken	
c5/p5- signal		Ready		Sleep	<b>p</b> 6	Running	1		
c6/p6- unlock		Ready		Sleep	pl	Running	1		
		Ready		Sleep	p2	Running	1		
	•	Ready		Sleep	<b>p</b> 3	Sleep	1	Must sleep (full)	
	C2	Running		Sleep		Sleep	1	Recheck conditio	
	× c4	Running		Sleep		Sleep	0	T <sub>c1</sub> grabs data	
	✓ c5	Running		Ready		Sleep	0	Oops! Woke Te2	





















COMPUTER BOOT SEQUENCE: OS WITH DIRECT EXECUTION								
OS		Program						
1. Create entr 2. Allocate me	ry for process list emory for							
Without <i>limits</i> or running programs, the OS wouldn't be in control of anything and would "just <u>be a library</u> "								
5. Clear regist 6. Execute cal	t <b>ers</b>   main()	<pre>7. Run main() 8. Execute return from main()</pre>						
9. Free memo 10. Remove fr	ory of process rom process list							
February 13, 2019	inter 2019] iology, University of Washington - Tacoma	L10.37						





