timestamp (tick)	BLOCKED	READY (in order)	current thread (and any transition)	comments / others
0	-	-	-	-
10	-	t1	main	t1 created and forking (adding to READY)
20	-	t1, t2	main	t1 forked t2 created and forking (adding to READY)
30	-	t1, t2, t3	main	t2 forked t3 created and forking (adding to READY)
40	-	t1, t2, t3, t4	main	t3 forked t4 created and forking (adding to READY)
50	-	t1, t2, t3, t4	main	t4 forked Executing Join(t4) // waiting for t4 before continuing main thread main releases semaphore Join
60	main	t2, t3, t4	main -> t1	main thread sleep (added to BLOCKED) switch to t1
70	main	t3, t4, t1	t1 -> t2	t1 yielded (added to READY) Context Switch to t2
80	main	t4, t1, t2	t2 -> t3	t2 yielded (added to READY) Context Switch to t3
90	main	t1, t2, t3	t3 -> t4	t3 yielded (added to READY) Context Switch to t4
100	main	t2, t3, t4	t4 -> t1	t4 yielded (added to READY) Context Switch to t1
110	main	t3, t4	t1 -> t2	t1.value++ and print (value = 1) Context Switch to t2 Delete t1 (Finish())
120	main	t4	t2 -> t3	t2.value-- and print (value = -1) Context Switch to t3 Delete t2 (Finish())
130	main	-	t3 -> t4	t3.value-- and print (value = -1) Context Switch to t4 Delete t3 (Finish())
140	main	-	t4	t4.value++ and print (value = 1) Cotinuing from Join(t4) in Tick 50 t4 acquires semaphore Join
150	-	main	t4	main thread added to READY t4 releases semaphore in_join
160	-	-	t4 -> main	t4 sleep (added to BLOCKED) Context Switch to main Delete t4 (Finish()) main acquires semaphore in_join
170	-	-	main -> NULL	Check value End main (Finish()) End program as no more threads