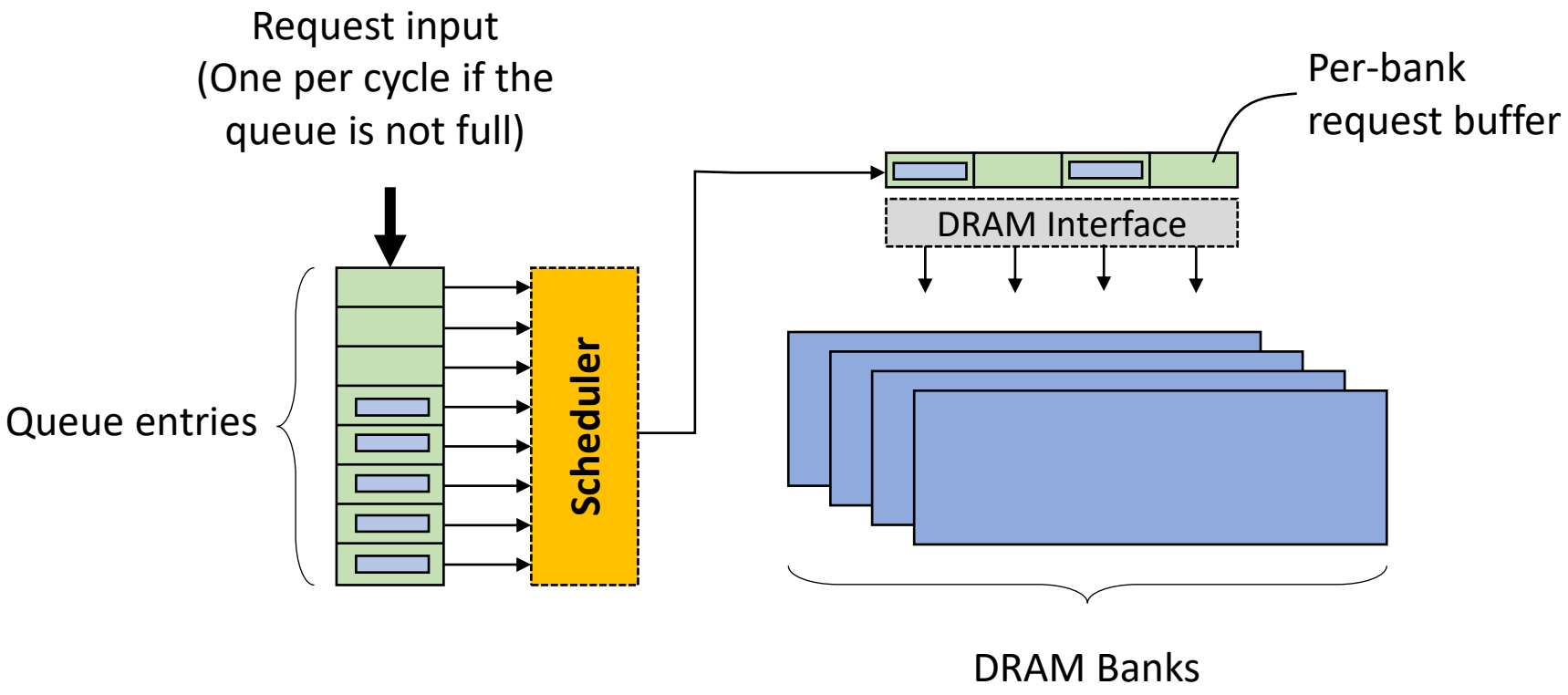
HW 2&3 Memory Scheduler (FCFS, FR-FCFS, and PARBS)

http://acm.cs.nthu.edu.tw/problem/11013/

http://acm.cs.nthu.edu.tw/problem/11030/

Simulated Architecture



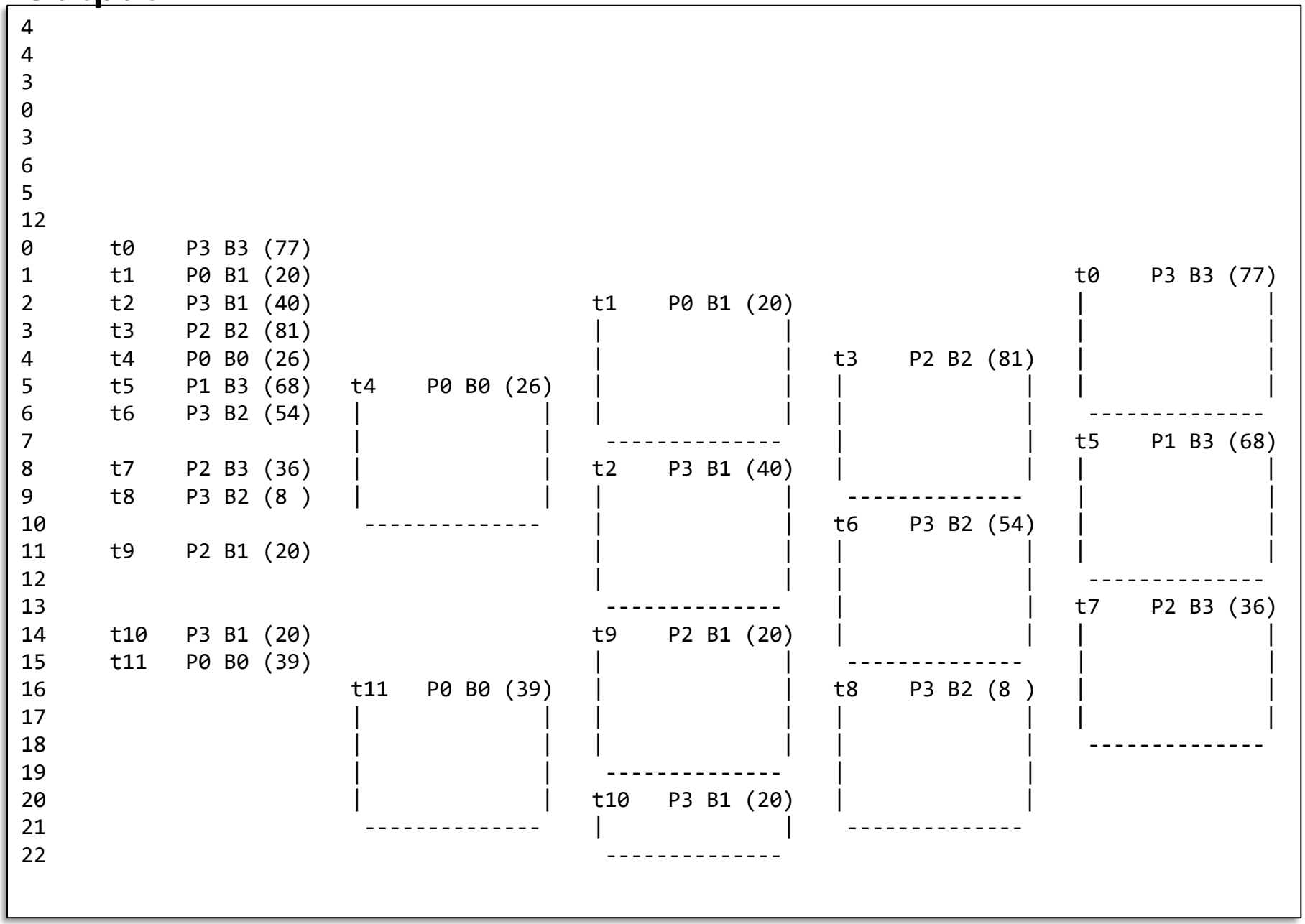
Input

4	Number of process (for PAR-BS)
4	Number of banks
3	Queue size
0	Policy (0:FCFS, 1:FR-FCFS, 2: PAR-BS)
3	Row hit latency
6	Row miss latency
5	Marking cap (for PAR-BS)
12	Number of the following requests
0 3 3 77	
1 0 1 20	
2 3 1 40	
3 2 2 81	
4 0 0 26	
5 1 3 68	
6 3 2 54	
7 2 3 36	
8 3 2 8	
9 2 1 20	
10 3 1 20	
11 0 0 39	

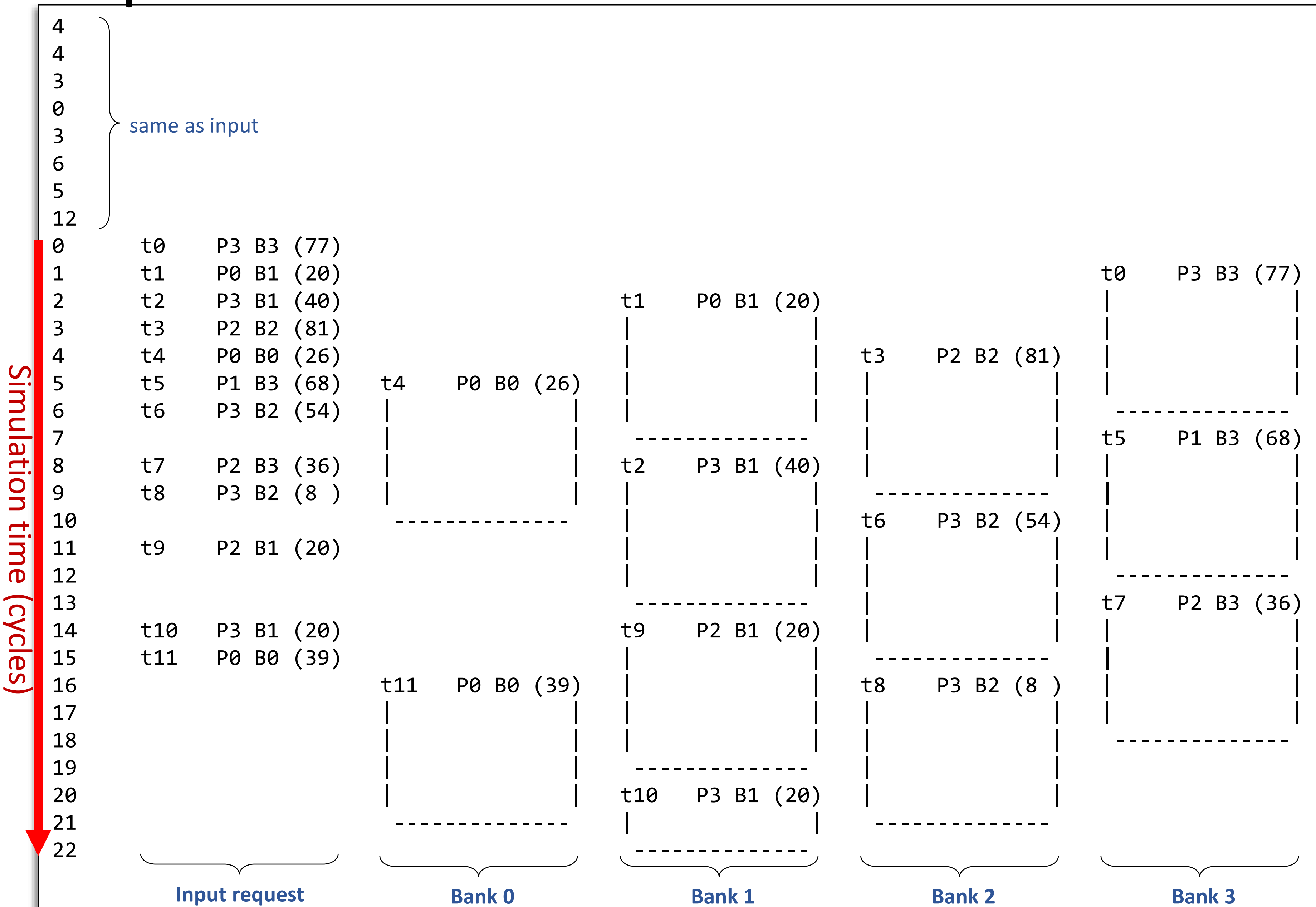
the process/thread id of the request is 0
the request is to bank 1
the request is to row 20

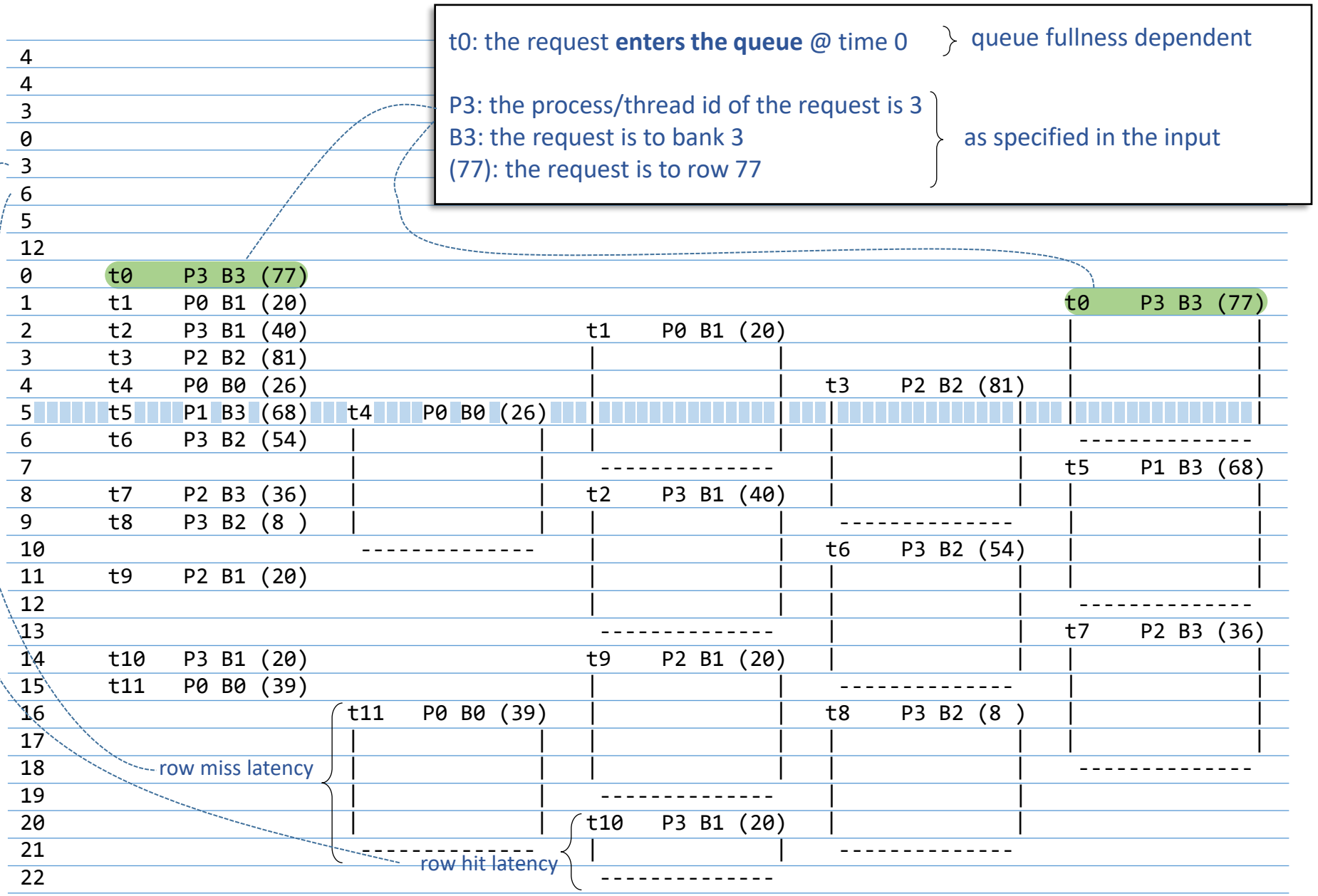
Serial number of request (not very important information)

Output

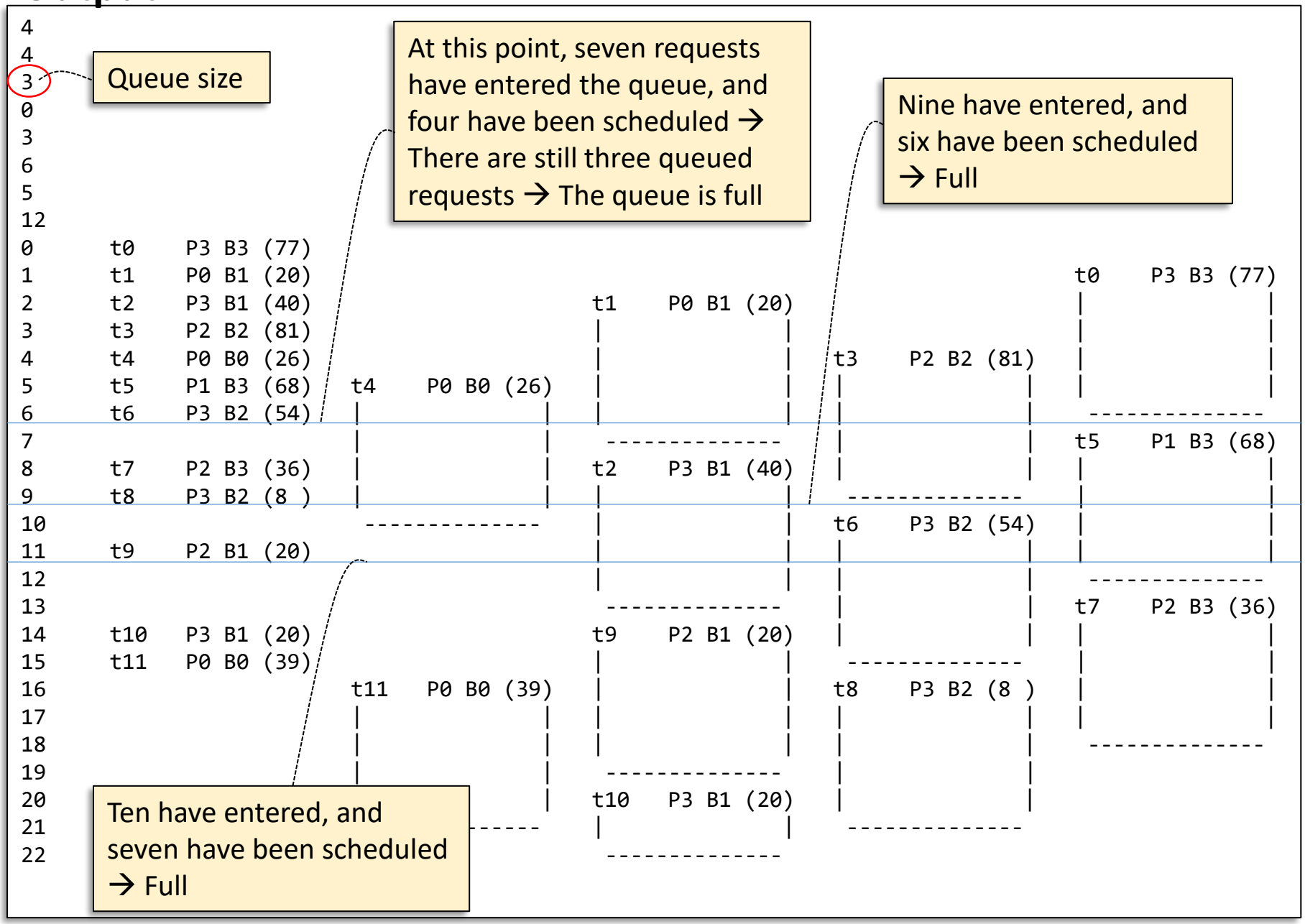

4
4
3
0
3
6
5
12
0 t0 P3 B3 (77)
1 t1 P0 B1 (20)
2 t2 P3 B1 (40)
3 t3 P2 B2 (81)
4 t4 P0 B0 (26)
5 t5 P1 B3 (68)
6 t6 P3 B2 (54)
7
8 t7 P2 B3 (36)
9 t8 P3 B2 (8)
10
11 t9 P2 B1 (20)
12
13
14 t10 P3 B1 (20)
15 t11 P0 B0 (39)
16
17
18
19
20
21
22
t4 P0 B0 (26)
t1 P0 B1 (20)
t2 P3 B1 (40)
t11 P0 B0 (39)
t9 P2 B1 (20)
t10 P3 B1 (20)
t3 P2 B2 (81)
t6 P3 B2 (54)
t8 P3 B2 (8)
t0 P3 B3 (77)
t5 P1 B3 (68)
t7 P2 B3 (36)

Output

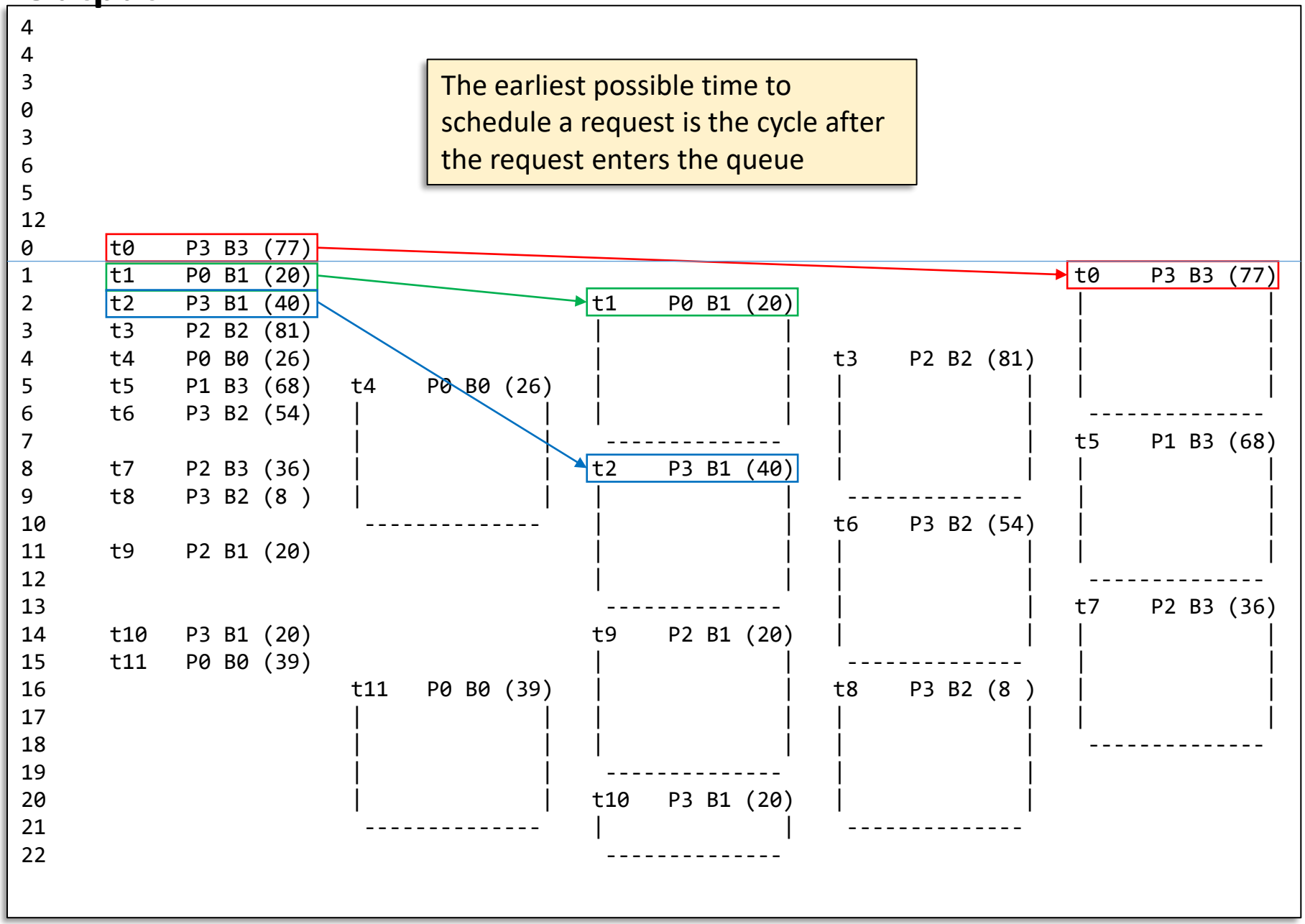

4
4
3
0
3
6
5
12
same as input
0 t0 P3 B3 (77)
1 t1 P0 B1 (20)
2 t2 P3 B1 (40)
3 t3 P2 B2 (81)
4 t4 P0 B0 (26)
5 t5 P1 B3 (68)
6 t6 P3 B2 (54)
7
8 t7 P2 B3 (36)
9 t8 P3 B2 (8)
10
11 t9 P2 B1 (20)
12
13
14 t10 P3 B1 (20)
15 t11 P0 B0 (39)
16
17
18
19
20
21
22
t4 P0 B0 (26)
t11 P0 B0 (39)
t1 P0 B1 (20)
t2 P3 B1 (40)
t9 P2 B1 (20)
t10 P3 B1 (20)
t3 P2 B2 (81)
t6 P3 B2 (54)
t8 P3 B2 (8)
t0 P3 B3 (77)
t5 P1 B3 (68)
t7 P2 B3 (36)
Simulation time (cycles)
Input request
Bank 0
Bank 1
Bank 2
Bank 3


t0: the request **enters the queue** @ time 0
queue fullness dependent
P3: the process/thread id of the request is 3
B3: the request is to bank 3
(77): the request is to row 77
as specified in the input
row miss latency
row hit latency

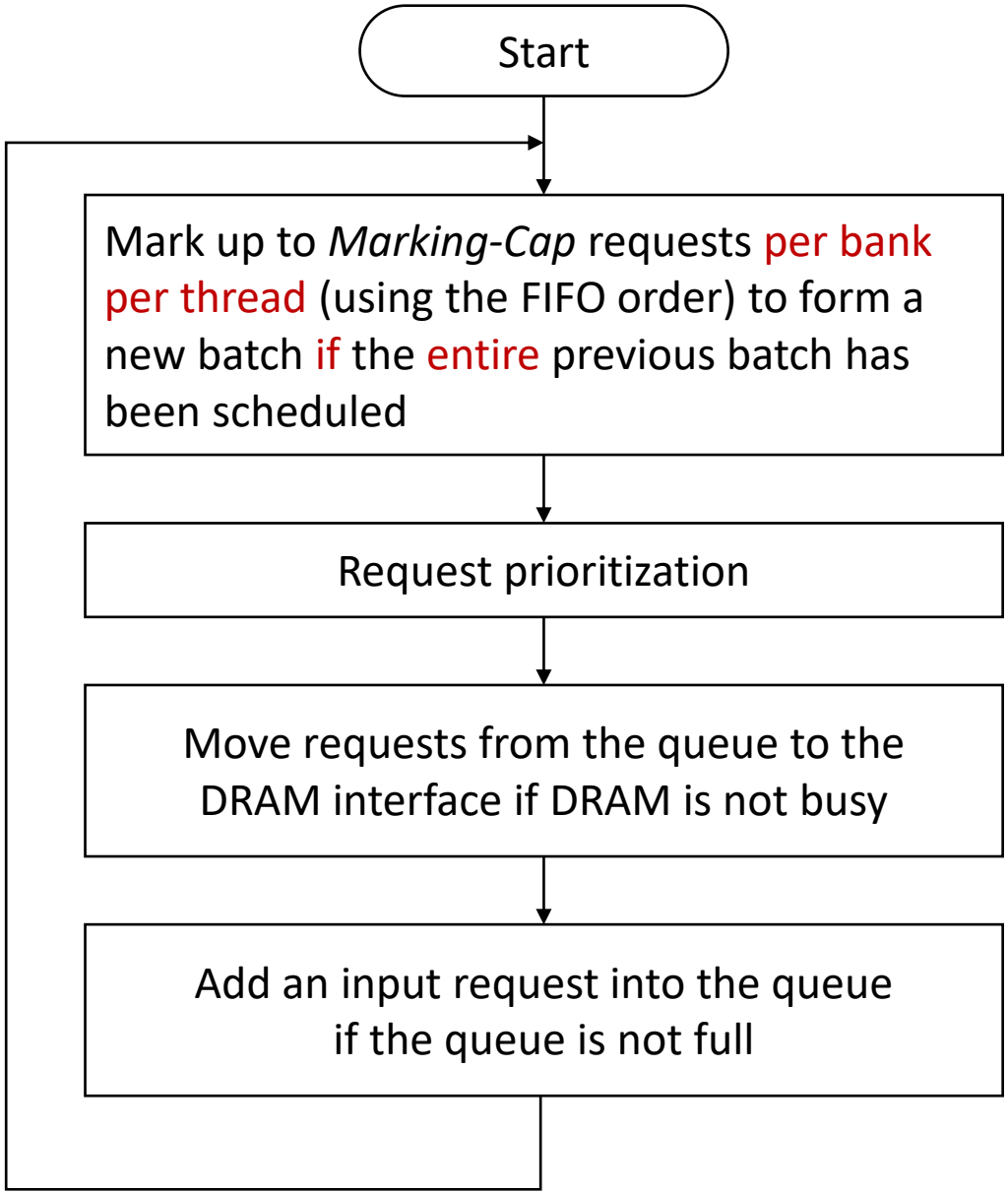
Output


4
4
3
0
3
6
5
12
0 t0 P3 B3 (77)
1 t1 P0 B1 (20)
2 t2 P3 B1 (40)
3 t3 P2 B2 (81)
4 t4 P0 B0 (26)
5 t5 P1 B3 (68)
6 t6 P3 B2 (54)
7
8 t7 P2 B3 (36)
9 t8 P3 B2 (8)
10
11 t9 P2 B1 (20)
12
13
14 t10 P3 B1 (20)
15 t11 P0 B0 (39)
16
17
18
19
20
21
22
Queue size
At this point, seven requests have entered the queue, and four have been scheduled → There are still three queued requests → The queue is full
Nine have entered, and six have been scheduled → Full
Ten have entered, and seven have been scheduled → Full

Output


4
4
3
0
3
6
5
12
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
The earliest possible time to schedule a request is the cycle after the request enters the queue
t0 P3 B3 (77)
t1 P0 B1 (20)
t2 P3 B1 (40)
t3 P2 B2 (81)
t4 P0 B0 (26)
t5 P1 B3 (68)
t6 P3 B2 (54)
t7 P2 B3 (36)
t8 P3 B2 (8)
t9 P2 B1 (20)
t10 P3 B1 (20)
t11 P0 B0 (39)

PARBS


Start
Mark up to *Marking-Cap* requests per bank per thread (using the FIFO order) to form a new batch if the entire previous batch has been scheduled
Request prioritization
Move requests from the queue to the DRAM interface if DRAM is not busy
Add an input request into the queue if the queue is not full

Request Prioritization

1: **BS—Marked-requests-first:** Marked ready requests are prioritized over requests that are not marked.
2: **RH—Row-hit-first:** Row-hit requests are prioritized over row-conflict/closed requests.
3: **RANK—Higher-rank-first:** Requests from threads with higher-rank are prioritized over requests from lower-ranked threads.
4: **FCFS—Oldest-first:** Older requests are prioritized over younger requests.

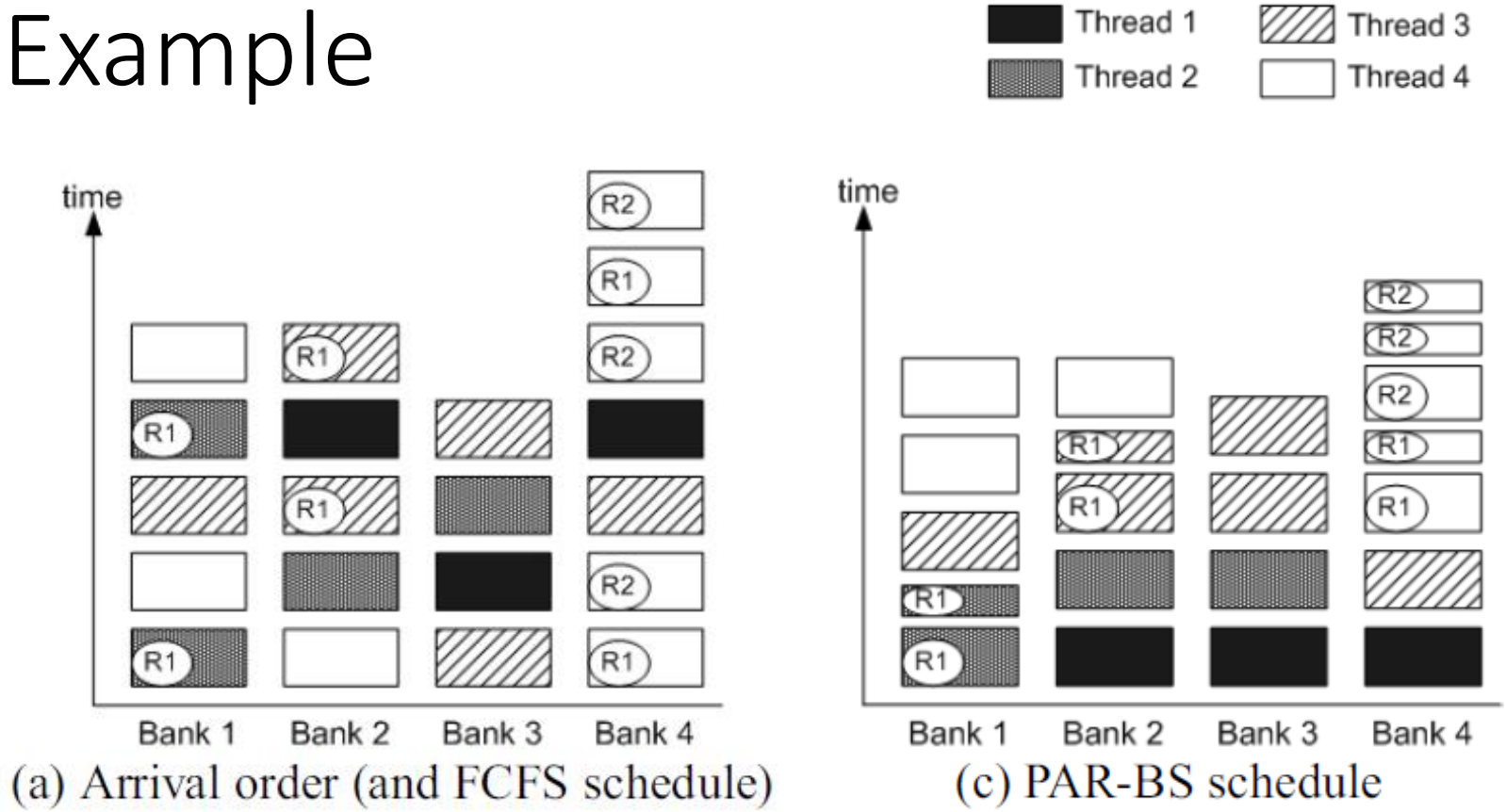
Request Ranking

1: **Max rule:** For each thread, the scheduler finds the maximum number of marked requests to any given bank, called max-bank-load. A thread with a lower max-bank-load is ranked higher than a thread with a higher max-bank-load.

2: **Tie-breaker Total rule:** For each thread, the scheduler keeps track of the total number of marked requests, called total-load. If threads are ranked the same based on the Max rule, a thread with a lower total-load is ranked higher than a thread with a higher total-load. ~~Any remaining ties are broken randomly.~~

3. Remaining ties are broken according to process/thread IDs.
P0 < P1 < P2 .. (larger ID → higher rank) and so on.

Example



	Max Bank Load
	1
	2
	2
	5

Input

4	Number of processes/threads
4	Number of banks
3	Queue size
0	Policy (0:FCFS, 1:FR-FCFS, 2: PAR-BS)
3	Row hit latency
6	Row miss latency
5	Marking cap
12	Number of the following requests
0 3 3 77	
1 0 1 20	
2 3 1 40	
3 2 2 81	
4 0 0 26	
5 1 3 68	
6 3 2 54	
7 2 3 36	
8 3 2 8	
9 2 1 20	
10 3 1 20	
11 0 0 39	

the process/thread id of the request is 0
the request is to bank 1
the request is to row 20

Serial number of request (not very important information)

Grading

- 100% Online Judge
  - http://acm.cs.nthu.edu.tw/problem/11013/
  - http://acm.cs.nthu.edu.tw/problem/11030/