

Program 3 Analysis

1. Briefly define a *queue* data structure.

A queue is a data structure that operates based on first in, first out. Elements are added at one end and come out the other.

What evidence did you find that a *queue* container was operating in the load balancer?

A queue load balancer just does the tasks in the order that they are received.

```
queue
Finished at 149 E pid 5/19 process time was 133
Finished at 222 C pid 3/25 process time was 206
Finished at 248 D pid 4/26 process time was 234
Finished at 257 C pid 7/26 process time was 32
Finished at 381 B pid 2/35 process time was 365
Finished at 425 A pid 1/39 process time was 405
Finished at 429 E pid 6/38 process time was 276
Finished at 513 D pid 8/50 process time was 262
Finished at 547 C pid 9/52 process time was 282
Finished at 638 D pid 13/60 process time was 120
Finished at 640 A pid 11/59 process time was 212
Finished at 646 B pid 10/58 process time was 261
Finished at 685 A pid 16/59 process time was 42
```

for example, the first task to be completed was the shortest one given, which was PID 5, and the first task given, the task with PID of 1 is completed further down, because it took a longer time to do.

2. Briefly define a *stack* data structure.

A stack operates based on first in, last out. Element are added and removed from the top of the stack.

What evidence did you find that a *stack* container was operating in the load balancer?

You can tell the load balancer uses a stack, because as you go down the list, the tasks that are being completed are always the newest ones. This is probably not a good container to use because since a stack is first in, last out, some tasks just never get completed and stay at the bottom of the stack--for example, the task with PID 1 was never completed during this run.

```

Finished at 9368 D pid 844/664 process time was 107
Finished at 9372 C pid 823/663 process time was 392
Finished at 9469 E pid 831/670 process time was 365
Finished at 9627 C pid 847/683 process time was 254
Finished at 9660 A pid 850/686 process time was 293
Finished at 9693 D pid 848/689 process time was 321
Finished at 9719 E pid 858/692 process time was 246
Finished at 9796 B pid 849/699 process time was 425
Finished at 9917 C pid 872/714 process time was 284
Finished at 9975 A pid 876/717 process time was 310

```

3. Briefly define a *priority queue* data structure.

A priority queue arranges elements based on some definition of priority that they have. Higher priority items go to the front of the queue..

What evidence did you find that a *Max priority queue based on process time* was operating in the load balancer?

```

priority queue max process time
Finished at 374 E pid 3/37 process time was 358
Finished at 393 D pid 5/38 process time was 378
Finished at 417 C pid 10/41 process time was 404
Finished at 431 B pid 9/40 process time was 415
Finished at 460 A pid 7/42 process time was 447
Finished at 857 C pid 17/81 process time was 436
Finished at 861 B pid 14/80 process time was 426
Finished at 869 E pid 12/79 process time was 490
Finished at 875 A pid 43/80 process time was 409
Finished at 883 D pid 21/79 process time was 487
Finished at 1339 E pid 59/122 process time was 469
Finished at 1341 B pid 84/121 process time was 476
Finished at 1343 D pid 56/120 process time was 457
Finished at 1345 A pid 93/119 process time was 464
Finished at 1357 C pid 71/119 process time was 497
Finished at 1782 C pid 89/163 process time was 422
Finished at 1808 D pid 104/164 process time was 463
Finished at 1810 A pid 94/163 process time was 462
Finished at 1816 B pid 106/163 process time was 470
Finished at 1824 E pid 121/163 process time was 483
Finished at 2272 C pid 158/195 process time was 489
Finished at 2275 A pid 182/194 process time was 463
Finished at 2281 B pid 170/193 process time was 459
Finished at 2284 E pid 157/192 process time was 458
Finished at 2293 D pid 166/193 process time was 475
Finished at 2732 C pid 207/233 process time was 458
Finished at 2735 A pid 183/232 process time was 455
Finished at 2739 E pid 161/231 process time was 451
Finished at 2741 B pid 194/230 process time was 454
Finished at 2768 D pid 222/234 process time was 466
Finished at 3220 A pid 246/273 process time was 483
Finished at 3222 C pid 233/272 process time was 486

```

You can see that the load balancer using a priority queue based on max processing time always completes tasks that take the longest first. The process time is consistently above 400, in the screenshot.

What evidence did you find that a *Min priority queue based on process time* was operating in the load balancer?

```

priority queue min process time
Finished at 103 D pid 7/8 process time was 88
Finished at 217 C pid 8/14 process time was 203
Finished at 291 B pid 9/23 process time was 270
Finished at 331 B pid 30/29 process time was 35
Finished at 375 A pid 10/31 process time was 355
Finished at 409 E pid 6/31 process time was 391
Finished at 563 D pid 13/43 process time was 456
Finished at 579 E pid 41/44 process time was 166
Finished at 682 C pid 20/51 process time was 460
Finished at 731 B pid 37/55 process time was 396
Finished at 761 B pid 69/56 process time was 25
Finished at 778 D pid 54/55 process time was 213
Finished at 797 C pid 64/56 process time was 112
Finished at 825 A pid 40/57 process time was 447
Finished at 893 D pid 70/62 process time was 112
Finished at 914 E pid 56/65 process time was 332
Finished at 917 C pid 73/64 process time was 114
Finished at 1120 A pid 75/82 process time was 294
Finished at 1132 C pid 84/83 process time was 213
Finished at 1246 B pid 71/90 process time was 480
Finished at 1288 D pid 81/90 process time was 391
Finished at 1349 E pid 85/95 process time was 430
Finished at 1360 A pid 104/94 process time was 239
Finished at 1395 A pid 120/96 process time was 33
Finished at 1462 C pid 106/101 process time was 325
Finished at 1488 D pid 115/102 process time was 198
Finished at 1498 D pid 132/102 process time was 6

```

For the min priority queue based on process time, you can see that the load balancer is taking care of tasks with small process times as it receives them.

What evidence did you find that a *Max priority queue based on process ID* was operating in the load balancer?

```
priority queue max pid
Finished at 80 A pid 9/13 process time was 69
Finished at 96 B pid 1/14 process time was 82
Finished at 101 B pid 20/14 process time was 4
Finished at 112 C pid 5/13 process time was 96
Finished at 118 D pid 4/12 process time was 96
Finished at 224 E pid 10/20 process time was 206
Finished at 244 E pid 22/20 process time was 19
Finished at 250 A pid 17/19 process time was 169
Finished at 289 E pid 31/21 process time was 43
Finished at 321 B pid 2/25 process time was 219
Finished at 331 B pid 39/25 process time was 7
Finished at 347 C pid 8/25 process time was 234
Finished at 350 A pid 23/24 process time was 95
Finished at 358 D pid 13/24 process time was 236
Finished at 399 E pid 27/27 process time was 98
Finished at 431 B pid 28/31 process time was 98
Finished at 439 E pid 44/30 process time was 35
Finished at 458 D pid 42/30 process time was 96
Finished at 462 C pid 37/29 process time was 110
Finished at 465 A pid 34/28 process time was 111
Finished at 531 B pid 48/29 process time was 95
Finished at 569 E pid 51/30 process time was 129
Finished at 618 D pid 26/33 process time was 155
Finished at 642 C pid 30/34 process time was 179
Finished at 665 A pid 35/35 process time was 195
Finished at 706 B pid 54/38 process time was 173
Finished at 728 D pid 59/37 process time was 105
Finished at 774 E pid 45/41 process time was 200
Finished at 797 C pid 57/44 process time was 154
Finished at 806 B pid 68/44 process time was 96
Finished at 810 A pid 63/43 process time was 142
Finished at 837 C pid 77/44 process time was 36
Finished at 871 B pid 78/46 process time was 58
```

For this implementation, it is similar to a normal queue in that tasks are just completed in the order they're created.