Daniel Maxson

OS ps #3

10/28/2015

#1.

|  |  |  |
| --- | --- | --- |
|  | Allocated | Max |
| P0 | 2,0,1,0 | 3,2,1,1 |
| P1 | 1,1,0,0 | 1,2,0,2 |
| P2 | 1,1,0,0 | 1,1,2,0 |
| P3 | 1,0,1,0 | 3,2,1,0 |
| P4 | 0,1,0,1 | 3,1,0,1 |

Total [6,4,2,2] Available [1,1,0,1]

P0 is not safe because we cannot fulfill the max request of 2 of resource R1. [3,**1**,1,1] < [3,2,1,1]

P1 is not safe because we cannot fulfill the max request of 2 of resource R3. [1,2,0,**1**] < [1,2,0,2]

P2 is not safe because we cannot fulfill the max request of 2 of resource R2. [1,1,**0**,1] < [1,1,2,0]

P3 is not safe because we cannot fulfill the max request of 3 of resource R0, nor can we fulfill the max request of resource R1. [**2**,**1**,1,0] < [3,2,1,0]

P4 is not safe because we cannot fulfill the max request of 3 of resource R0. [**1**,1,0,1] < [3,1,0,1]

No process can proceed without releasing resources.

#2.

No philosopher can acquire 2 chopsticks because they are all holding and waiting. P1 can serve themselves but then not eat.

Total [5,2] Available [0,1]

For all P0-P4 [**1**,1] < [2,1] assuming they only need one serving spoon.

#3.

FCFS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 |

0 10 11 13 14 19

Wait time 0+10+11+13+14 = 48. 48/5 = 9.6s. TAT 19-0+14-0+13-0+11-0+10-0 = 67. 67/5 = 13.4s.

SJF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P2 | P4 | P3 | P5 | P1 |

0 1 2 4 9 19

Wait time 0+1+2+4+9 = 16. 16/5 = 3.2s. TAT 19-0+9-0+4-0+2-0+1-0 = 35. 35/5 = 7s

Priority

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P2 | P5 | P1 | P3 | P4 |

0 1 6 16 18 19

Wait time 0+1+6+16+18 = 41. 4118-0+16-0+6-0+1-0 /5 = 8.2s. TAT 19-0+= 60. 60/5 = 12s.

RR

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 | P1 | P3 | P5 | P1 | P5 |

0 1 2 3 4 5 6 7 8 9 10

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P1 | P5 | P1 | P5 | P1 | P1 | P1 | P1 | P1 |

10 11 12 13 14 15 16 17 18 19

P2 finishes at 2s, P4 finished at 4s, P3 finishes at 7s, P5 finishes at 14s, and P1 finishes at 19s.

Wait time 1+1+9+9 = 20. 20/5 = 4s. TAT 2-0+4-0+7-0+14-0+19-0 = 46. 46/5 = 9.2s.

SJF has the lowest average wait and TAT time.

#4. Time owed

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | T1 | T2 | T3 | T4 | T5 |
| Q1 | -0.8s | 0.2s | 0.2s | 0.2s | 0.2s |
| Q2 | -0.6s | -0.8s | 0.4s | 0.4s | 0.4s |
| Q3 | -.04s | -0.6s | -0.6s | 0.6s | 0.6s |
| Q4 | -0.2s | -0.4s | -0.4s | -0.4s | 0.8s |
| Q5 | 0 | -0.2s | -0.2s | -0.2 | -0.2s |

T1 has wait time of 0s, T2 has wait time of 1s, T3 has wait time of 2s, T4 has wait time of 3s, and T5 has wait time of 4s.

B. At the level of tasks CFS seems to be fairly allocating time to each task on a single processor system. It processes a portion of one task while making the other tasks wait to receive their fair share of processing time. Process P1 happens to execute faster and first in this process and sub task model, so while the tasks are fairly processed the processes can run faster / be processed and finished if they have less subtasks to complete. P1 has only two tasks and would seem intuitive that it should take less time to process than a process that has 3 tasks.

Seems to be a multi-level combination of SJF and RR.