6.14

a. $\alpha=0$ $\tau_0=100ms$

$\tau_1=\alpha t_0+(1-\alpha)\tau_0=\tau_0=100ms$ s.t. the length we predict as next CPU burst is nothing to do with the actual length of previous CPU bursts i.e. the predicted length is a constant value.

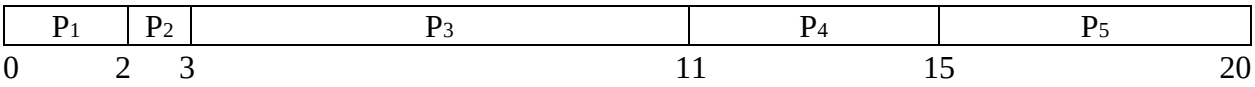
b. $\alpha=0.99$ $\tau_0=10ms$

$\tau_1=\alpha t_0+(1-\alpha)\tau_0=0.99t_0+0.01\tau_0\approx t_0$ s.t. the length we predict as next CPU burst has strong relation to the actual length of the nearest previous CPU burst

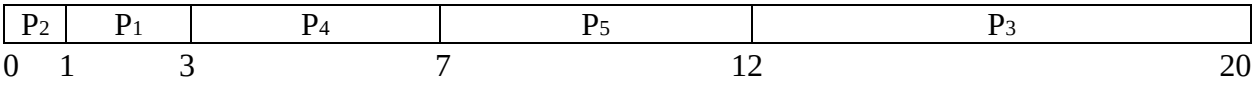
6.16

a. Gantt charts

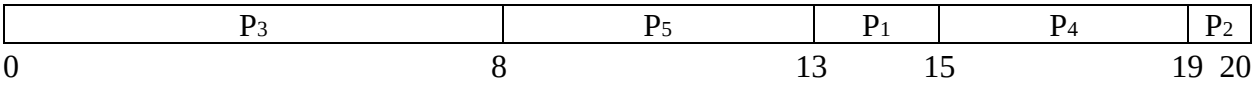
FCFS



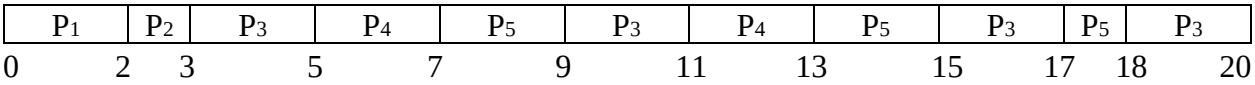
SJF



nonpreemptive priority



RR (quantum = 2)



b. turnaround time

FCFS: P_1:2 P_2: 3 P_3: 11 P_4: 15 P_5: 20

SJF: P_1:3 P_2: 1 P_3: 20 P_4: 7 P_5: 12

nonpreemptive priority: P_1:15 P_2: 20 P_3: 8 P_4: 19 P_5: 13

RR: P_1:2 P_2: 3 P_3: 20 P_4: 13 P_5: 18

c. waiting time

FCFS: P_1:0 P_2: 2 P_3: 3 P_4: 11 P_5: 15

SJF: P_1:1 P_2: 0 P_3: 12 P_4: 3 P_5: 7

nonpreemptive priority: P_1:13 P_2: 19 P_3: 0 P_4: 15 P_5: 8

RR: P_1:0 P_2: 2 P_3: 12 P_4: 9 P_5: 13

d. average waiting time

FCFS: 6.2 SJF: 4.6 nonpreemptive priority: 11 RR: 7.2

10.11 2069, 1212, 2296, 2800, 544, 1618, 356, 1523, 4965, 3681

a. FCFS

2150 2069 1212 2296,2800 544 1618 356 1523 4965 3681

2150-2069+2069-1212+2296-1212+2800-2296+2800-544+1618-544+1618-356+1523-356+4965-1523+4965-3681=13011

b. SSTF

2150 2069 2296 2800 3681 4965 1618 1523 1212 544 356

2150-2069+2296-2019+2800-2296+3681-2800+4965-3681+4965-1618+1618-1523+1523-1212+1212-544+544-356=7586

c. SCAN

2150 2296 2800 3681 4965 4999 2069 1618 1523 1212 544 356

2296-2150+2800-2296+3681-2800+4965-3681+4999-4965+4999-2069+2069-1618+1618-1523+1523-1212+1212-544+544-356=7492

d. LOOK

2150 2296 2800 3681 4965 2069 1618 1523 1212 544 356

2296-2150+2800-2296+3681-2800+4965-3681+4965-2069+2069-1618+1618-1523+1523-1212+1212-544+544-356=7424

e. C-SCAN

2150 2296 2800 3681 4965 4999 0 356 544 1212 1523 1618 2069

2296-2150+2800-2296+3681-2800+4965-3681+4999-4965+4999+356+544-356+1212-544+1523-1212+1618-1523+2069-1618=9917

f. C-LOOK

2150 2296 2800 3681 4965 356 544 1212 1523 1618 2069

2296-2150+2800-2296+3681-2800+4965-3681+4965-356+544-356+1212-544+1523-1212+1618-1523+2069-1618=9137

10.17 – when computing, choose the most optimal way. Remember that parity uses modulo-2 additions (e.g. 1,1,0,1 would have yield a parity of 1 (odd parity), whereas 1,1,0,0 would yield a parity of 0 (even)

a. A write of one block of data

Read the old data of the block;
Read the parity block;
Calculate new parity and write the parity block;
Write the block.
Totally 2 blocks are accessed for 4 times.

b. A write of seven continuous blocks of data

Write the first 4 blocks and calculate the first parity block;
Write the first parity block;
Write the next 3 blocks;
Read the eighth block;
Calculate the second parity block and write this block.
Totally 4+1+3+1+1=10 blocks are accessed.

11.9

The new file will be regarded as the old file by the existed link to that file.

Delete the file and all the links to this file or not delete the file until all the links to this file are deleted.

11.14

Fetch these blocks of file in advance to reduce the time occupied by I/O when the application is running.