Brandon Patton

Homework 2

1. The busy waiting solution using turn variable in the textbook solves the mutual exclusion problem when two processes are running on a shared-memory multiprocessor. This is because each process takes turns handing off control as demonstrated by the following trace:
    a. turn initially set to 0
    b. process 0 begins as turn = 0, enters critical region
    c. process 0 enters non-critical region, turn now set to 1
    d. process 1 begins as turn = 1, enters critical region
    e. process 1 finishes critical region, enters non-critical region, turn now set to 0
    f. process 0 enters critical region (since turn = 0), set turn = 1
    g. both processes now in their non-critical regions
    h. process 0 cannot re-enter its critical region, waiting on process 1

    Since each process depends on the other finishing, control is traded off fairly, but the one process cannot progress without the other handing off control. Therefore, mutual exclusion is solved but busy waiting still occurs.

2.
    a. Round Robin – each process is given an equal amount of quantum to process. Assumed quantum = 1
        i. A: 30 minutes, B: 23 minutes, C: 8 minutes, D: 17 minutes, E: 28 minutes
        ii. Mean process turnaround time = (30 + 23 + 8 + 17 + 28)/5 = **21.2 minutes**
    b. Priority Scheduling – highest priority finishes first, then in order of priorities
        i. B: 6 Minutes, E: 14 minutes, A: 24 minutes, C: 26 minutes, D: 30 minutes
        ii. Mean process turnaround time = (6 + 14 + 24 + 26 + 30)5 = **20 minutes**
    c. First-Come, First-Served – Finish jobs in order they arrive
        i. A: 10 Minutes, B: 16 minutes, C: 18 minutes, D: 22 minutes, E: 30 minutes
        ii. Mean process turnaround time = (10 + 16 + 18 + 22 + 30)/5 = **19.2 minutes**
    d. Shortest Job First – Finish shortest job first, then in ascending order of process time
        i. C: 2 minutes, D: 6 minutes, B: 12 minutes, E: 20 minutes, A: 30 minutes
        ii. Mean process turnaround time = (2 + 6 + 12 + 20 + 30)/5 = **14 minutes**

3.
    a. $35/50+20/100+10/200+x/250 \le 1$
    $700+200+50+4x \le 1000$
    $4x \le 50$
    x = **12.5 msec**

4.

current resource allocation matrix C:

	R1	R2	R3	R4	R5
P1	0	1	1	1	2
P2	0	1	0	1	0
P3	0	0	0	0	1
P4	2	1	0	0	0

request matrix R:

	R1	R2	R3	R4	R5
P1	1	1	0	2	1
P2	0	1	0	2	1
P3	0	2	0	3	1
P4	0	2	1	1	0

Existing Resources E(24144): R1 = 2, R2 = 4, R3 = 1, R4 = 4, R5 = 4

Available Resources A(01021): R1 = 0, R2 = 1, R3 = 0, R4 = 2, R5 = 1

1st row of R not less than or equal to A

2nd row of R = (01021), equal to A = (01021), therefore add 2nd row of C to A

A = (02031), P2 finished

3rd row of R = (02031), equal to A = (02031), therefore add 3rd row of C to A

A = (02032), P3 finished

1st and 4th row of R not less than or equal to A at this stage in the process, algorithm terminates and processes P1 and P4 are deadlocked.

5.

	Allocated	Maximum	Available
Process A	10211	11212	00x11
Process B	20110	22210	
Process C	11010	21310	
Process D	11110	11221	

Needed resources:

Process A (01001)

Process B (02100)

Process C (10300)

Process D (00111)

--Process D needs 00111 from 00x11

try x = 1

Process D finishes, 11221 now available

--Process A needs 01001 from 11221

Process A finishes, 21432 now available

--Process C needs 10300 from 21432

Process C finishes, 32442 now available

--Process B needs 02100 from 32442

Process B finishes, 52552 now available

Smallest value of x = 1, safe sequence: D, A, C, B

I pledge my honor that I have abided by the Stevens Honor System

Brandon Patton