Cüneyt EREM

21202398

Cs-342

Homework2

1)

Output;
10
20
20
30
30
30
30

2)

	CPU time	period
A:	10	80
B:	20	60
C:	40	150

For the EDF, there is a calculation;

a :	80	80-160	160-240	240-320	
b :	60	60-120	120-180	180-240	240-300
c :	150	150-300	300-450		

Here is the EDF table first 300 unit time;

b	a	c	b	c	a	empty	b	empty
0-20	20-30	30-60	60-80	80-90	90-100	100-120	120-140	140-150

c	a	c	b	c	empty	b	a	empty
150-160	160-170	170-180	180-200	200-220	220-240	240-260	260-270	270-300

3)
Head tail
Cpu: N, N-1, N-2, ... ,1
Exe time: N, N-1, N-2, ... ,1

Avg waiting time:
A: FCFS: (0 + N + (N + N-1) + (N + N-1 + N-2) + ... + (N + N-1 + ... 3 + 2))/N

B: SJF: (0 + 1 + (1 + 2) + (1 + 2 + 3) + ... + (1 + 2 + ... + N-1))/N

C: RR(q = 1): (N-1 + (N-1 + N-2) + ... + (N-1 + N-2 + ... + 1))/N

Max Response time:
A: FCFS: (N + N-1 + N-2 + ... + 2) = N(N+1)/2 -1

B: SJF: (N-1 + N-2 + ... + 1) = N(N-1)/2

C: RR(q = 1): N-1

4)

```
Sempahore TobaccoAndPaper = PaperAndMatches = MatchesAndTobacco = 0;
Semaphore SmokingDone = SmokingDone2 = SmokingDone3 = 1;

void agent()
{
          while(true) {

    int random = rand() % 3;

    if(random == 0) {
                        signal(TobaccoAndPaper);
        break;
           }
           else if(random == 1) {
                        signal(PaperAndMatches);
        break;
           }
           else {
                        signal(MatchesAndTobacco);
        break;
           }
         }
}

void Smoker1()
{
    while(true) {

        wait(TobaccoAndPaper);
                        wait(SmokingDone);

        smoke();

        signal(SmokingDone);
    }
}
```

```
void Smoker2()
{
    while(true) {

        wait(PaperAndMatches);
                        wait(SmokingDone2);

        smoke();

        signal(SmokingDone2);
    }
}

void Smoker3()
{
    while(true) {

        wait(MatchesAndTobacco);
                        wait(SmokingDone3);

        smoke();

        signal(SmokingDone3);
    }
}
```

5)

WT: waiting time, FT: finished time, TAT: turned around time

A)

FCFS	WT	FT	TAT
A	0	40	40
B	25	65	50
C	40	95	70
D	60	140	105
E	85	165	110

B)

SJF	WT	FT	TAT
A	0	40	40
B	25	65	50
C	65	120	95
D	85	165	130
E	10	90	65

C)

RR(Q=10)	WT	FT	TAT
A	40	80	80
B	65	105	90
C	70	125	100
D	85	165	130
E	70	150	95

D)

RR(Q=30)	WT	FT	TAT
A	55	95	95
B	15	55	40
C	30	85	60
D	85	165	130
E	70	150	95

6)

```
Monitor MonitorAllocX{
        boolean busy = FALSE;
        condition x;
        int count = N;

        void request() {
                if (count == 0 && busy == TRUE)
                        x.wait();
                busy = TRUE;
        }
        void release() {
                busy = FALSE;
                x.signal();
        }
}
```

7)

System call, invocation system is a concept that lots of actions can be performed by kernel to provide services to user process. In the code, function of the C library gets this service from the kernel. For example, POSIX defines the interface for programs and function for the program. It can use these functions.

8)

-The kernel performs thread creation, scheduling, and management in kernel space.
-The kernel is managing the threads, if a thread performs a blocking system call.
-A multiprocessor environment, the kernel can schedule threads on different processors.
-Including Windows NT, Windows 2000, Solaris 2, BeOS, and Tru64 UNIX (formerlyDigital UN1X)-support kernel threads.