AMITY INTERNATIONAL SCHOOL

NAME Ishita Yadav

CLASS B6 SEC.

ASSIGNMENT NO.

SUBJECT OSSP ROLL NO. 20102168

DATE. T. SIGN.

## Assignment 2

1.

| Process | BT | Priority | |
|---|---|---|---|
| P1 | 10 | 3 | 9 8765 |
| P2 | 1 | 1 | ✓ |
| P3 | 2 | 3 | 1 ✓ |
| P4 | 1 | 4 | ✓ |
| P5 | 5 | 2 | 4 3 2 1 ✓ |

FCFS :

| P1 | P2 | P3 | P4 | P5 | |
|---|---|---|---|---|---|
| 0 | 10 | 11 | 13 | 14 | 19 |

SJF :

| ~~P3~~ P2 | P4 ~~P3~~ | P3 | P5 | P1 | |
|---|---|---|---|---|---|
| 0 | 1 | 2 | 4 | 9 | 19 |

Non Preemptive Priority :

| P2 | P5 | P1 | P3 | P4 | |
|---|---|---|---|---|---|
| 0 | 1 | 6 | 16 | 18 | 19 |

RR : q = 1

| P1 | P2 | P3 | P4 | P5 | P1 | P3 | P5 | P1 | P5 | P1 | P5 | P1 | P5 | P1 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 19 |

| ~~P1~~ | Turnaround Time | | | | Waiting Time | | | |
|---|---|---|---|---|---|---|---|---|
| | FCFS | SJF | Priority | RR | FCFS | SJF | Priority | RR |
| ~~P2~~ P1 | 10 | 19 | 16 | 19 | 0 | 9 | 6 | 9 |
| ~~P3~~ P2 | 11 | 1 | 1 | 2 | 10 | 0 | 0 | 1 |
| ~~P4~~ P3 | 13 | 4 | 18 | 7 | 11 | 2 | 16 | 5 |
| ~~P5~~ P4 | 14 | 2 | 19 | 4 | 13 | 1 | 18 | 3 |
| P5 | 19 | 9 | 6 | 14 | 14 | 4 | 1 | 9 |

NAME

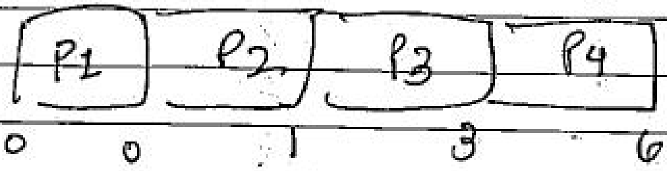
CLASS SEC. ASSIGNMENT NO.

SUBJECT ROLL NO. DATE. T. SIGN.

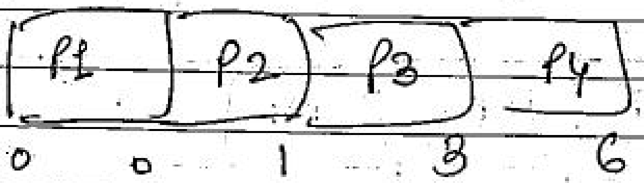
2.

| Process | BT | Priority | |
|---|---|---|---|
| P1 | 0 | 8 | |
| P2 | 1 | 4 | |
| P3 | 2 | 9 | 1 |
| P4 | 3 | 5 | 2 |

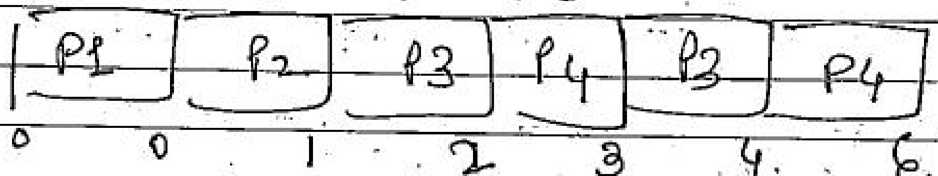
FCFS :



Preemptive : SJF



RR : q=1



| | Turnaround Time | | | Waiting Time | | |
|---|---|---|---|---|---|---|
| | FCFS | Pr. SJF | RR | FCFS | Pr. SJF | RR |
| P1 | 0 | 0 | 0 | 0 | 0 | 0 |
| P2 | 1 | 1 | 1 | 0 | 0 | 0 |
| P3 | 3 | 3 | 4 | 1 | 1 | 2 |
| P4 | 6 | 6 | 6 | 3 | 3 | 3 |
| Avg: | 2.5 | 2.5 | 2.75 | 1 | 1 | 1.25 |

3.

| Process | AT | BT | Priority | |
|---|---|---|---|---|
| P1 | 0 | 6 | 4 | 3 ✓ |
| P2 | 3 | 5 | 2 | |
| P3 | 3 | 3 | 6 | ✓ |
| P4 | 5 | 5 | 3 | ✓ |

Preemptive :

| % | P2, P3 | P4 | | | |
|---|---|---|---|---|---|
| P1 | P3 | P1 | P4 | P2 | |
| 0 | 3 | 6 | 9 | 14 | 19 |

Non Preemptive :

| | | | | |
|---|---|---|---|---|
| P1 | P3 | P4 | P2 | |
| 0 | 6 | 9 | 14 | 19 |

(4)

| Process | BT | | Waiting Time FCFS | SJF | RR |
|---|---|---|---|---|---|
| P1 | 10 | | 0 | 10 | 0 |
| P2 | 29 | 1st | $ 10 | 32 | 32 |
| P3 | 3 | ✓ | 39 | 0 | 20 |
| P4 | 7 | | 42 | 3 | 23 |
| PC | 12 | & | 49 | 20 | 40 |
| | | | ⇒ 28 | ⇒ 13 | ⇒ 23 |

FCFS :

| | | | | | |
|---|---|---|---|---|---|
| P1 | P2 | P3 | P4 | P5 | |
| 0 | 10 | 39 | 42 | 49 | 61 |

Non Pr. SJF

| | | | | | |
|---|---|---|---|---|---|
| P3 | P4 | P1 | P5 | P2 | |
| 0 | 3 | 10 | 20 | 32 | 61 |

RR q=10

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| P1 | P2 | P3 | P4 | P5 | P2 | P5 | P2 | |
| 0 | 10 | 20 | 23 | 30 | 40 | ~~49~~50 | 52 | 61 |

⇒ Non Preemptive SJF Gives least avg. waiting time

NAME Ishita Yadav

CLASS BC SEC.

ASSIGNMENT NO.

SUBJECT OSSP ROLL NO. 20103168

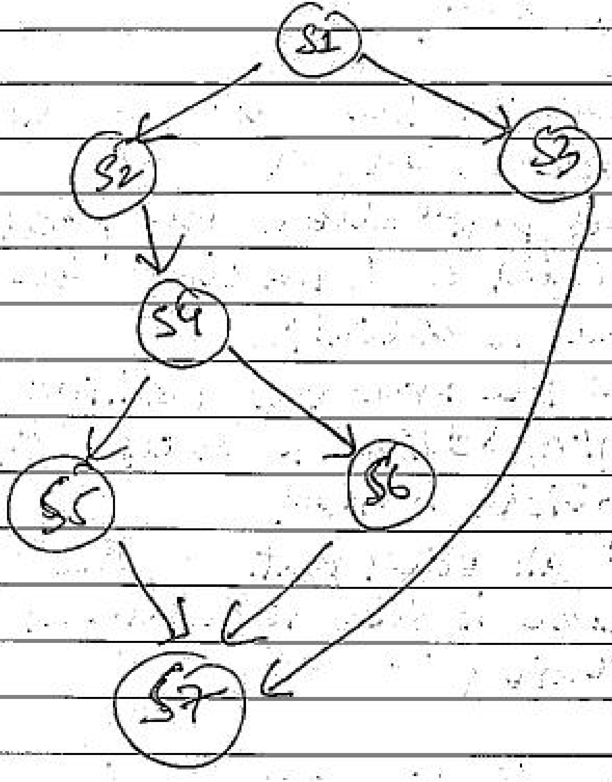
DATE. T. SIGN.

## Assignment : 3

1. 
- Entry Section : decides the entry of a particular process
- Critical Section : this part allows one process to enter and modify the shared variable
- Exit Section : allows the processes waiting in the Entry Section to enter into the critical section
- Remainder Section : all other parts of the code is the remainder section

2. Busy Waiting means that a process is waiting for a condition to be satisfied in a tight loop without relinquishing the processor. Alternatively, a process could wait by relinquishing the processor, and block on a condition and wait to be awakened at some inappropriate time in the future.
Busy waiting can be avoided but incurs the overhead associated with putting a process to sleep and having to wake it up when the appropriate program state is reached.

(2)



Initial Semaphors :

| | |
|---|---|
| sem 1 | (1) |
| sem 2 | (0) |
| sem 3 | (0) |
| sem 4 | (0) |
| sem 5 | (0) |

```
while (1) {
    wait (sem1)
        S1
    signal (sem2)
}
```

```
while (1) {
    wait (sem2)
        S2
    signal (sem3)
}
```

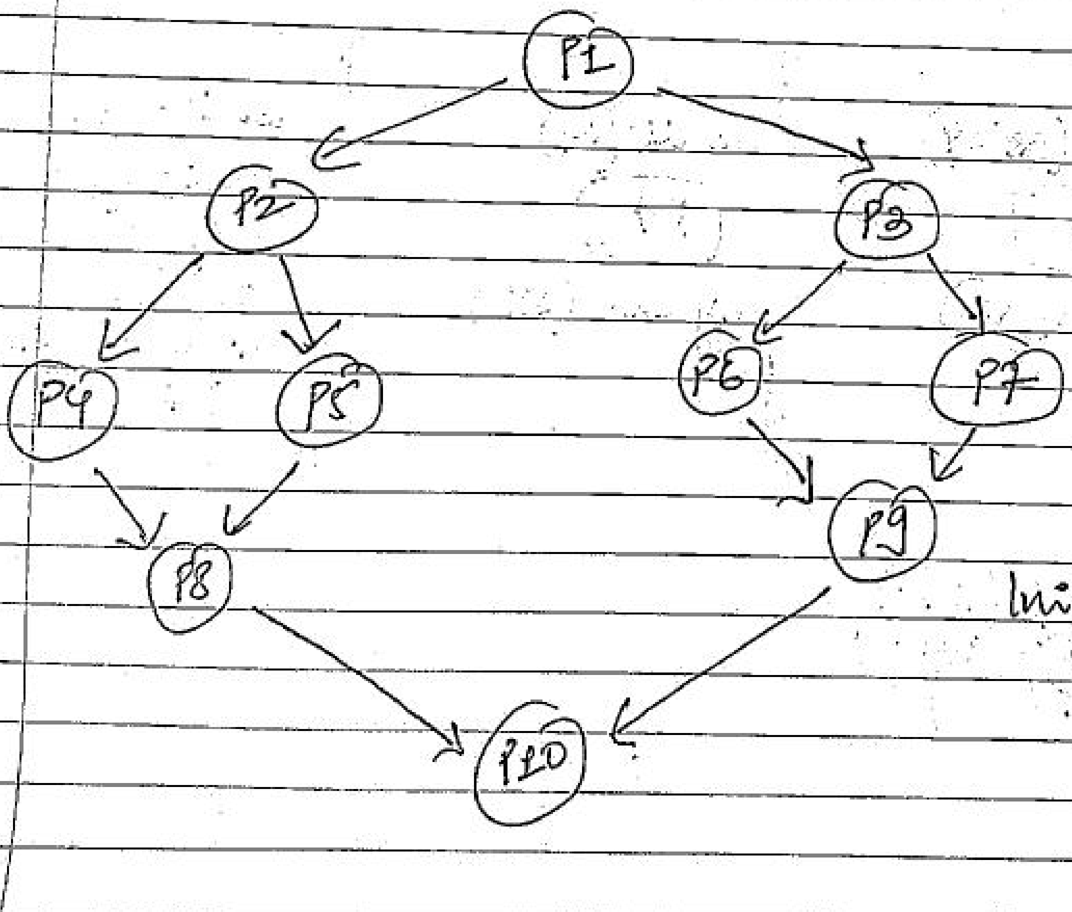
```
while (1) {
wait (sem2)
    S3
signal (sem4)
}
```

```
while (1) {
    wait (sem3)
        S4
    signal (sem5)
}
```

```
while (1) {
    wait (sem5)
      [ S5 ]
    signal (sem4)
}
```

```
while (1) {
    wait (sem5)
      [ S6 ]
    signal (sem4)
```

```
while (1) {
    wait (sem4)
      [ S7 ]
    ~~signal (sem4)~~
}
```



Initial Semaphores:
~~S1(1)~~ S1(1), S6(0)
S2(0), S7(0)
S3(0), S8(0)
S4(0), S9(0)
S5(0), S10(0)

```
{
    wait (S1)
    P1
    signal (S2)
}
```

```
{
    ~~sig~~ wait (S2)
    P2
    signal (S3)
}
```

```
{
    wait (S2)
    P3
    signal (S4)
}
```

```
{
    wait (S3)
    P4
    signal (S5)
}
```

```
{
    wait (S3)
    P5
    signal (S6)
}
```

```
{
    wait (S5)
    wait (S6)
    P8
    signal (S7)
}
```

```
{
    wait (~~S8~~ S4)
    P6
    signal (S8)
}
```

```
{
    wait (S4)
    P7
    signal (S9)
}
```

```
{
    wait (S8)
    wait (S9)
    P9
    signal (S10)
}
```

```
{
    wait (S7)
    wait (S10)
    P10
}
```

NAME____________________

CLASS________ SEC.________ ASSIGNMENT NO.________

SUBJECT________ ROLL NO.________ DATE________ T. SIGN.________

(4.) File sort utility, is often-used as a filter in a pipe
ls -l | sort would have sorted the lines of the output from ls. sort ~~would~~ reads the output of ls since it was not given any specific filename to read from & is reading from the standard input stream, which is connected to the output stream of ls via the pipe
ls - l | sort > sort.txt attempts to store the sorted list in sort.txt which is impossible.

(5.) A pipe usually connects only 2 processes, although any no. of child processes can be connected to each other and their related parent by a single pipe. So yes, children and grandchildren can use a pipe

~~(6)~~ (7) A race condition occurs when a device or system attempts to perform 2 or more operations at the same time, but because of the nature of the system, the operations must be done in the proper sequence
The presence of a race condition is considered a bad design because invalid executions and software bugs can occur due to this

NAME Ishita Yadav

CLASS B6 SEC.

SUBJECT OSSP ROLL NO. 20103168

ASSIGNMENT NO.

DATE. T. SIGN.

## Assignment : 4

(1.)

| Paging | Segmentation |
|---|---|
| ◦ Program is divided into fixed sized pages | ◦ Program is divided into variable size section. |
| ◦ OS is accountable | ◦ Compiler is accountable |
| ◦ Page size is determined by hardware | ◦ Section size is given by user |
| ◦ Faster | ◦ Slower |
| ◦ Internal fragmentation | ◦ External fragmentation |

(2)

| Segment | Base | Length |
|---|---|---|
| 0 | 330 | 124 |
| 1 | 876 | 211 |
| 2 | 111 | 99 |
| 3 | 498 | 302 |

(a.) 0, 99

↳ offset

Limit of 0 is 124

Range ⇒ [0, 124-1] = [0, 128]

Physical Address : 330 + 99 = 429

(b.) 2, 78

Physical Address : 111 + 78 = 189

(c.) 1, 265

↳ offset out of range

↳ segment fault

NAME

CLASS SEC. ASSIGNMENT NO.

SUBJECT ROLL NO. DATE T. SIGN.

(d) 3, 222

→ Physical Address : 222 + 498 = 720

(e) 0, 111

→ Physical Address : 111 + 330 = 440

(3) Physical Memory : $2^{24}$

LAS : $256 = 2^8$

Page Size : $2^{10}$

Logical Address Space = $2^8 \times 2^{10} = 2^{18}$ bytes

→ 18 bit address is required

(4) Virtual Address Space : $8 \times 2^{29} = 2^{32}$

No. of pages in each segment : $\frac{2^{29}}{2^8} = 2^{21}$

Offset within the page : 8 bits

(5) (a) LRU

~~1 2 3 4 2 1 5 6 2 1 2 3 7 6~~

3 frames

| | 1 | 2 | 3 | 4 | 2 | 1 | 5 | 6 | 2 | 1 | 2 | 3 | 7 | 6 | 3 | 2 | 1 | 2 | 3 | 6 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| frame1 | 1 | | | 4 | | | 5 | | | 1 | | | 7 | | | 2 | | | | |
| frame2 | | 2 | | | | | | 6 | | | | 3 | | | | | | | | |
| frame3 | | | 3 | | | 1 | | | 2 | | | | | 6 | | | 1 | | | 6 |

⇒ 15 faults

4 frames

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| f1 | 1 | | | | | | | | | | | | | 6 | | | | | | |
| f2 | | 2 | | | | | | | | | | | | | | | | | | |
| f3 | | | 3 | | | | 5 | | | | | 3 | | | | | | | | |
| f4 | | | | 4 | | | | 6 | | | | | 7 | | | | 1 | | | |

⇒ 10 faults

NAME

CLASS SEC.

SUBJECT ROLL NO.

ASSIGNMENT NO.

DATE T. SIGN.

| | | 1 | 2 | 3 | 4 | 2 | 1 | 5 | 6 | 2 | 1 | 2 | 3 | 7 | 6 | 3 | 2 | 1 | 2 | 3 | 6 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 frames | F1 | 1 | | | | | | | | | | | | | | | | | | | |
| | F2 | | 2 | | | | | | | | | | | | | | | | | | |
| | F3 | | | 3 | | | | | 6 | | | | | | | | | | | | |
| | F4 | | | | 4 | | | | | | | | 3 | | | | | | | | |
| | F5 | | | | | | | 5 | | | | | | 7 | | | | | | | |

⇒ 8 faults

| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 6 frames | F1 | 1 | | | | | | | | | | | | | | | | | | | |
| | F2 | | 2 | | | | | | | | | | | | | | | | | | |
| | F3 | | | 3 | | | | | | | | | | | | | | | | | |
| | F4 | | | | 4 | | | | | | | | | | | | | | | | |
| | F5 | | | | | | | 5 | | | | | | 7 | | | | | | | |
| | F6 | | | | | | | | 6 | | | | | | | | | | | | |

⇒ 7 faults (fewest)

(b.) FIFO

| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 frames | F1 | 1 | | | 4 | | | | 6 | | | | 3 | | | | 2 | | | | 6 |
| | F2 | | 2 | | | | 1 | | | 2 | | | | 7 | | | | 1 | | | |
| | F3 | | | 3 | | | | 5 | | | 1 | | | | 6 | | | | | 3 | |

⇒ 16 faults

| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 frames | F1 | 1 | | | | | | 5 | | | | | 3 | | | | | 1 | | | |
| | F2 | | 2 | | | | | | 6 | | | | | 7 | | | | | | 3 | |
| | F3 | | | 3 | | | | | | 2 | | | | | 6 | | | | | | |
| | F4 | | | | 4 | | | | | | 1 | | | | | | 2 | | | | |

⇒ 14 faults

| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 5 frames | F1 | 1 | | | | | | | 6 | | | | | | | | | | | | |
| | F2 | | 2 | | | | | | | | 1 | | | | | | | | | | |
| | F3 | | | 3 | | | | | | | | 2 | | | | | | | | | |
| | F4 | | | | 4 | | | | | | | | 3 | | | | | | | | |
| | F5 | | | | | | | 5 | | | | | | 7 | | | | | | | 7 |

⇒ 10 faults

NAME

CLASS SEC.

ASSIGNMENT NO.

SUBJECT ROLL NO.

DATE T. SIGN.

6 frames

| | 1 | 2 | 3 | 4 | 2 | 1 | 5 | 6 | 2 | 1 | 2 | 3 | 7 | 6 | 3 | 2 | 1 | 2 | 3 | 6 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| f1 | 1 | | | | | | | | | | | | 7 | | | | | | | |
| f2 | | 2 | | | | | | | | | | | | | | | 1 | | | |
| f3 | | | 3 | | | | | | | | | | | | | | | 2 | | |
| f4 | | | | 4 | | | | | | | | | | | | | | | 3 | |
| f5 | | | | | | | 5 | | | | | | | | | | | | | |
| f5 | | | | | | | | 6 | | | | | | | | | | | | |

⇒ 10 faults

(c) optimal

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| f1 | 1 | | | | | | | | | | | 3 | | | | | | | | |
| f2 | | 2 | | | | | | | | | | | 7 | | | 2 | | | | |
| f3 | | | 3 | 4 | | | 5 | 6 | | | | | | | | | 1 | | | 6 |

⇒ 11 faults

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| f1 | 1 | | | | | | | | | | | | 7 | | | | 1 | | | |
| f2 | | 2 | | | | | | | | | | | | | | | | | | |
| f3 | | | 3 | | | | | | | | | | | | | | | | | |
| f4 | | | | 4 | | | 5 | 6 | | | | | | | | | | | | |

⇒ 8 faults

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| f1 | 1 | | | | | | | | | | | | | | | | | | | |
| f2 | | 2 | | | | | | | | | | | | | | | | | | |
| f3 | | | 3 | | | | | | | | | | | | | | | | | |
| f4 | | | | 4 | | | | | | | | | 7 | | | | | | | |
| f5 | | | | | | | 5 | 6 | | | | | | | | | | | | |

⇒ 7 faults.

NAME Ishita Yadav

CLASS B6 SEC.

SUBJECT OSSP ROLL NO. 20103168

ASSIGNMENT NO.

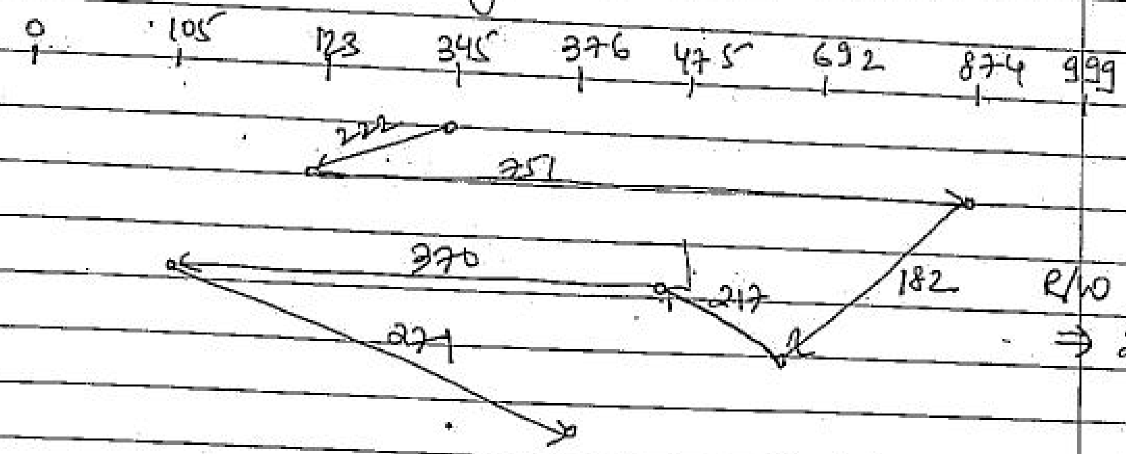
DATE. T. SIGN.

## Assignment: 5

(1)

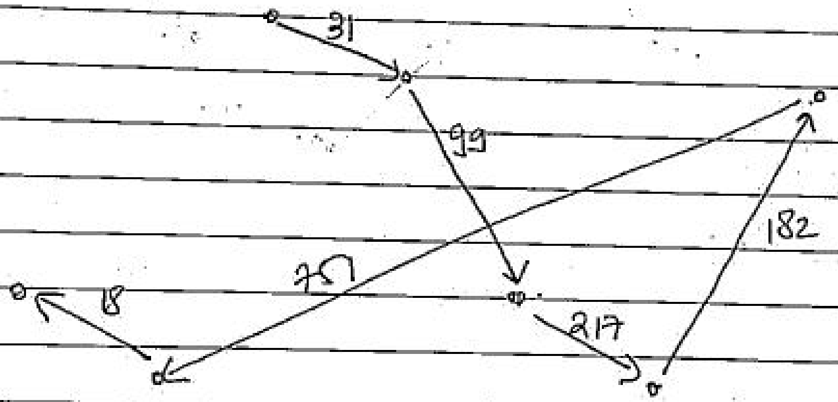
FCFS:

~~FCFS~~:



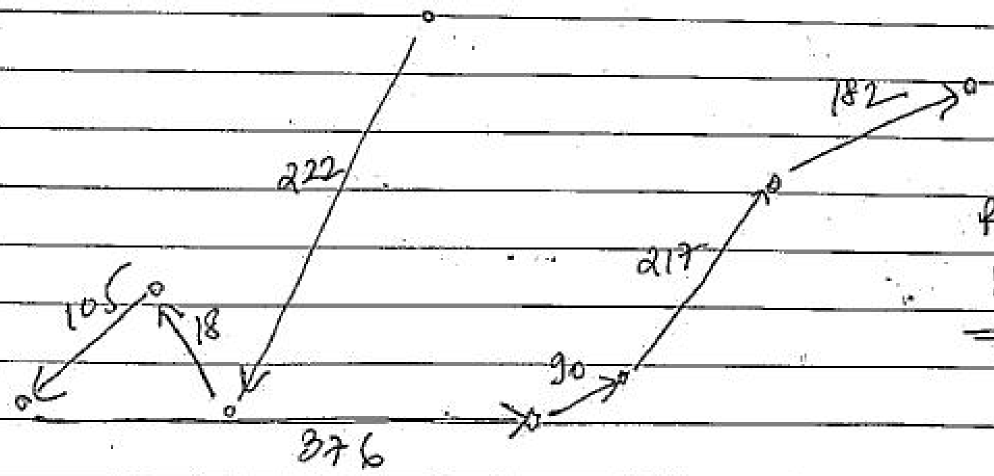
R/W Movement ⇒ 2013 tracks

SSTF:



R/W Head Movement ⇒ 1298 tracks

SCAN:



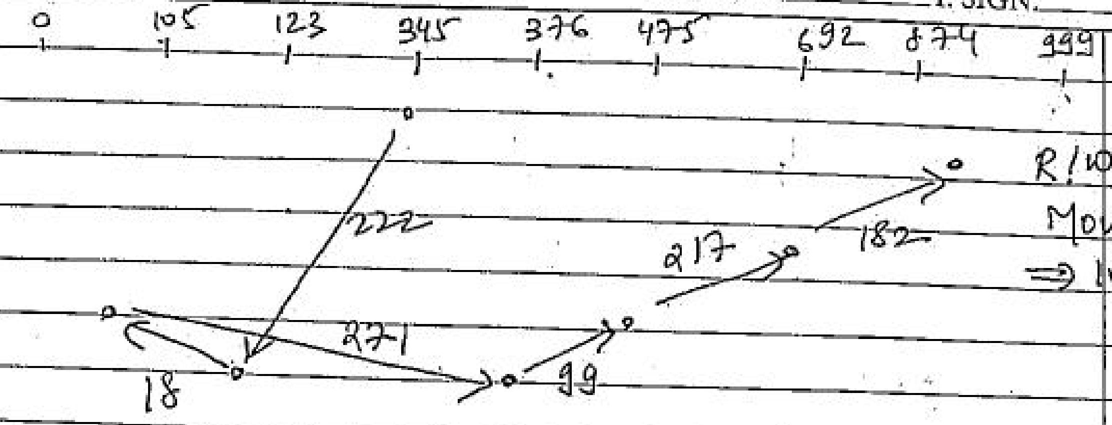
R/W Head Movement ⇒ 1219 tracks

NAME

CLASS SEC. ASSIGNMENT NO.

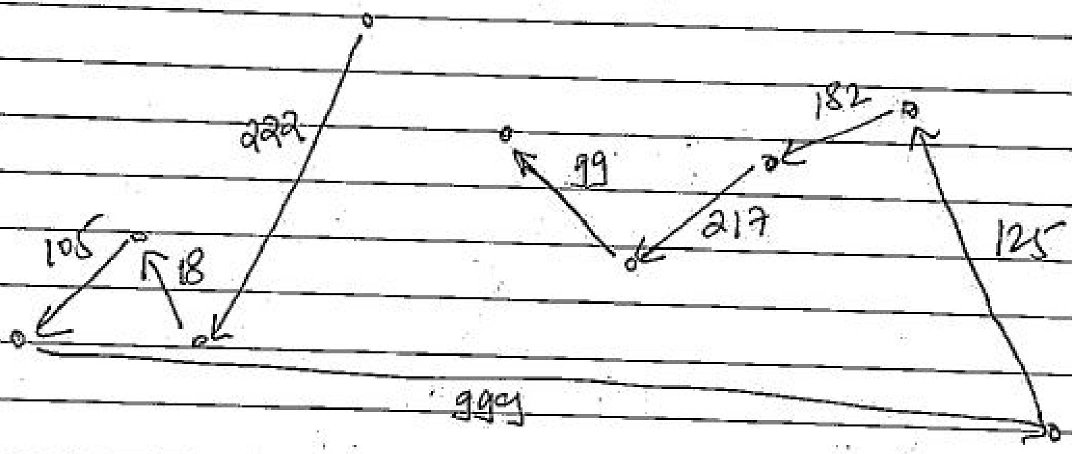
SUBJECT ROLL NO. DATE. T. SIGN.

LOOK:



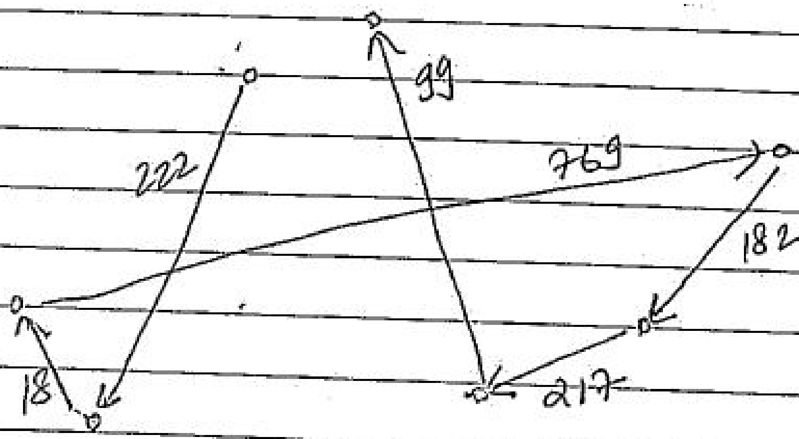
R/W Head Movement ⇒ 1009 tracks

CSCAN :



R/W Head Movement ⇒ 1967 tracks

CLOOK:



R/W Head Movement ⇒ 1507 tracks

AMITY INTERNATIONAL SCHOOL

NAME

CLASS SEC. ASSIGNMENT NO.

SUBJECT ROLL NO. DATE T. SIGN.

| (2) | Next request to be serviced | No. of cylinders travelled | Average seek length |
|---|---|---|---|
| FCFS : | 55 | 45 | |
| | 58 | 3 | |
| | 39 | 19 | $\frac{498}{9} = 55.3$ |
| | 18 | 21 | |
| | 90 | 72 | |
| | 160 | 70 | |
| | 150 | 10 | |
| | 38 | 112 | |
| | 184 | 146 | |
| SSTF | 90 | 10 | |
| | 58 | 32 | |
| | 55 | 3 | |
| | 39 | 16 | 27.5 |
| | 38 | 1 | |
| | 18 | 20 | |
| | 150 | 132 | |
| | 160 | 10 | |
| | 184 | 24 | |
| SCAN | 150 | 50 | |
| | 160 | 10 | |
| | 184 | 24 | |
| | 90 | 94 | 27.8 |
| | 58 | 32 | |
| | 55 | 3 | |
| | 39 | 16 | |
| | 38 | 1 | |
| | 18 | 20 | |

CSCAN

| | | |
|---|---|---|
| 150 | 50 | |
| 160 | 10 | 35.6 |
| 184 | 24 | |
| 18 | 166 | |
| 38 | 20 | |
| 39 | 1 | |
| 55 | 16 | |
| 58 | 3 | |
| 90 | 32 | |

(3) 1 track → 1 Revolution ⟹ 11.11 ms
Seek time ⟹ (19456 - 1) x2 = 39 sec
Total = 3498 sec
⟹ 58 min

(4) 1024 x 16 x 63 x 512 = 528 million bytes
= 504 MB

5. (a) SSTF : Shortest Seek Time first. Selects ~~the~~ request which is closest to the current head position before moving the head away to service other requests.

(b.) FCFS : First Come First Serve Scheduling. Automatically executes the queued processes and requests in the order of their arrival. It allocates the job that first arrived in the queue to the CPU, then 2nd and so on.

(c) SCAN : head starts from one end of the disk and moves towards the other end, servicing requests in between one by one and reach the other end. Then the direction is reversed & the process continues as head continuously scans back and forth to access the disk.

(d) CSCAN : modified version of SCAN disk that services the requests more uniformly. Moves the head from one end servicing all the requests to the other end. However, as soon as the head reaches the other end, it immediately returns to the beginning of the disk and starts servicing again.