POSSESSION OF MOBILES IN EXAMS IS A UFM PRACTICE.

Name_______________ **Enrolment No.______________**

Jaypee Institute of Information Technology, Noida
End Semester Examination 2022
Semester – 5th

Course Title: Operating System and Systems Programming **Max. Hours: 2Hr**
Course Code: 15B11CI412 **Max. Marks: 35**

	Attempt all questions
1.	Consider the following set of processes, with arrival time and execution time reported in milliseconds, along with the priority of each process. Note that Pi has higher priority than Pj if **Priority (Pi) > Priority (Pj):** **[Strict Marking 2 Marks per Algorithm]** **[For Each Algorithm: 1 Mark Gantt Chart + .5 Marks Avg Waiting Time+ .5 Avg Turn Around Time]**

Process	Arrival Time	Execution Time	Priority
P0	0	5	5
P1	1	2	1
P2	2	1	2
P3	3	4	3
P4	2	4	0
P5	5	3	4

Calculate the average turnaround time and average waiting time for the following scheduling algorithms.

- Round Robin.
- Non-Pre-emptive Priority Scheduling (processes are scheduled in the priority order)
- Shortest Remaining Time First

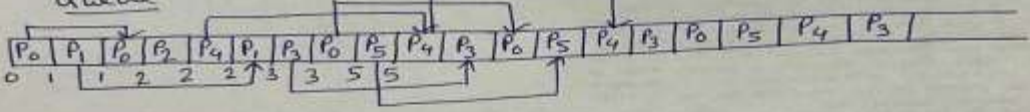
Assume a scheduling quantum of 1 millisecond. Also, draw a timeline (called a Gantt Chart) illustrating the schedule.

[CO-2][6 MARKS]

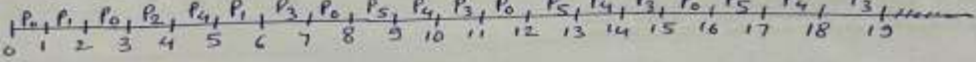
Round Robin

Q.3 solution

Queue



GANTT



$TAT = (16-0) + (6-1) + (4-2) + (19-3) + (18-2) + (17-5)$

$= 16 + 5 + 2 + 16 + 16 + 12 = 67$

$\boxed{\text{Avg } TAT = 67/6 = 11.16}$

$WT = (16-5) + (5-2) + (2-1) + (16-4) + (16-4) + (12-3)$

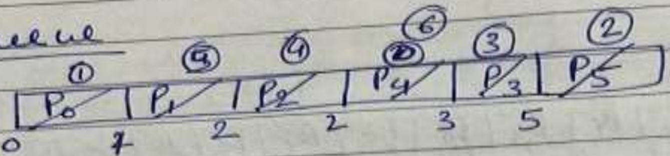
$= 11 + 3 + 1 + 12 + 12 + 9 = 48$

$\boxed{\text{Avg } WT = 48/6 = 8}$

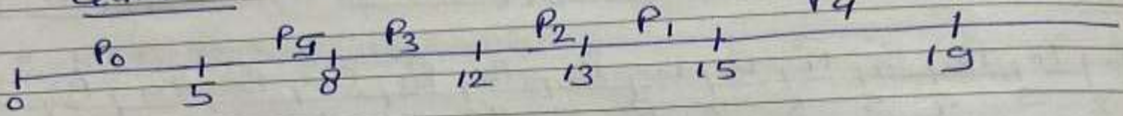
Priority Scheduling

Priority

Queue



Gantt



$TAT = CT - AT$

$= (5-0) + (15-1) + (13-2) + (12-3) + (19-2)$
$+ (8-5)$

$= 5 + 14 + 11 + 9 + 17 + 3$

$= 59$

Average $\boxed{TAT = 59/6 = \cancel{11.8}\ 9.83}$

$WT = (5-5) + (14-2) + (11-1) + (9-4) + (17-4) + (3-3)$

$= 0 + 12 + 10 + 5 + 13 + 0 = 40$

$\boxed{\text{Avg } WT = 40/6 = 6.66}$

SHORTEST REMAINING TIME FIRST

SRTF

Queue

P0	P1	P2	P4	P3	P5
0	1	2	2	3	5

Gantt

P0	P1	P2	P0	P0	P5	P4	P3

0 1 3 4 5 8 11 15 19

$$TAT = CT - AT$$
$$= (8-0) + (3-1) + (4-2) + (19-3) + (15-2) + (11-5)$$
$$= 8 + 2 + 2 + 16 + 13 + 6$$
$$= 47$$

$$\boxed{\text{Average } TAT = 47/8 = \cancel{9.4}\ 7.83}$$

$$WT = (8-5) + (2-2) + (2-1) + (16-4) + (13-4) + (6-3)$$
$$= 3 + 0 + 1 + 12 + 9 + 3 = 28$$

$$\boxed{\text{Avg } WT = 28/6 = 4.66}$$

2. Consider a virtual memory system that translates addresses using a translation look-aside buffer (TLB) and a two-level page table that resides in the main memory. Each access to the main memory takes 200 ns, while the search in the TLB takes 40 ns. It takes 3 microseconds for each page to be read from or written to the secondary storage. Assume that the hit ratio for the TLB is 85%, and that the page fault rate is 20%. A dirty page must be written back to secondary storage before the needed page can be read from secondary storage, which accounts for 30% of all page faults. TLB update time is insignificant. Determine the average memory access time.

[CO-3][5 MARKS]

SOL	**[5 MARKS FOR CORRECT ANSWER**] [**STRICT MARKING**] Two-level paging Main memory access time= 200 ns TLB access time= 40 ns Secondary memory access time= 3 microsecond = 3000 ns TLB hit =85%, Page fault rate= 20%, dirty page= 30% So, Average memory access time= .85(40+200)+.15[.80(40+200+200+200)+.20[.70(40+200+200+200+3000)+.30(40+200+200+200+3000+3000]] =417 **Only Formula =2 Marks** **Formula + Calculation=3 Marks** **Formula+ calculation+right answer=5 Marks**
3	Consider the following page reference. Indicate page faults and calculate the total number of page faults for FIFO, optimal, and LRU Page replacement algorithms. The total number of available frames is 4. **1,2,3,2,5,6,3,4,6,3,7,3,1,5,3,6,3,4,2,4,3,4,5,1** **[CO-3][4 MARKS]**
SOL	**[Flexible Marking, Required complete execution diagram and correct no of page fault]** **[FIFO—1 Marks,** **Optimum and LRU------1.5 MARKS Each]**

FIFO
.5 Marks for diagram
.5 Marks For Page Faults

FIFO

f_4					5	5	5	5	5	5	5	3	3	3	3	3	3	4	4	4	4	4	4	1
f_3			3	3	3	3	3	3	3	3	7	7	7	7	7	6	6	6	6	6	6	6	5	5
f_2		2	2	2	2	2	2	4	4	4	4	4	4	5	5	5	5	5	5	5	3	3	3	3
f_1	1	1	1	1	1	6	6	6	6	6	6	6	1	1	1	1	1	1	2	2	2	2	2	2
	X	X	X	H	X	X	H	X	H	H	X	X	X	X	H	X	H	X	X	H	X	H	X	X

Total no of page faults = 16

For **optimum and LRU**
1 Marks for diagram
.5 for page faults

Optimal 1, 2, 3, 2, 5, 6, 3, 4, 6, 3, 7, 3, 1, 5, 3, 6, 3, 4, 2, 4, 3, 4, 5, 1

f_4					5	5	5	4	4	4	7	7	7	5	5	5	5	5	5	5	5	5	5	5
f_3			3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
f_2		2	2	2	2	6	6	6	6	6	6	6	6	6	6	6	6	4	4	4	4	4	4	4
f_1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	1
	X	X	X	H	X	X	H	X	H	H	X	H	H	X	H	H	H	X	X	H	H	H	H	X

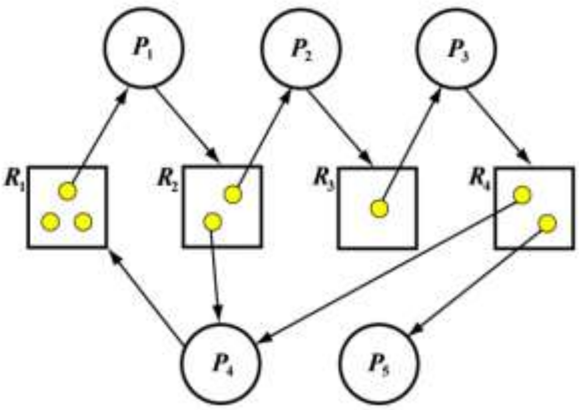
Total page fault = ~~7~~ 11

LRU 1, 2, 3, 2, 5, 6, 3, 4, 6, 3, 7, 3, 1, 5, 3, 6, 3, 4, 2, 4, 3, 4, 5, 1

f_4					5	5	5	5	5	5	7	7	7	7	7	6	6	6	6	6	6	6	5	5
f_3			3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
f_2		2	2	2	2	2	2	4	4	4	4	4	1	1	1	1	1	4	4	4	4	4	4	4
f_1	1	1	1	1	1	6	6	6	6	6	6	6	6	5	5	5	5	5	2	2	2	2	2	1
	X	X	X	H	X	X	H	X	H	H	X	H	X	X	H	X	H	X	X	H	H	H	X	X

Total page fault = 14

4. Consider the following resource allocating graph.



Do the following problems:

a) Convert the above diagram into the matrix representation (*i.e Allocation, request, and Available*) **[1.5 Marks for correct Diagram]**

	b) Write a deadlock detection algorithm. **[1.5 Marks for the correct algorithm, flexible marking]** c) Do a step-by-step execution of the deadlock detection algorithm. Write all intermediate computational values **[2 Marks for correct intermediate steps]** d) Is there a deadlock? If there is a deadlock, which processes are involved? **[1 Mark]** **[CO-2][6 MARKS]**
SOL **a)**	**[STRICK MARKING, ALL VALUES SHOULD BE CORRECT]**

Answer: The matrix representation of the given resource allocation graph is shown below:

	Allocation				*Request*				*Available*			
	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4
P_1	1	0	0	0	0	1	0	0	2	0	0	0
P_2	0	1	0	0	0	0	1	0				
P_3	0	0	1	0	0	0	0	1				
P_4	0	1	0	1	1	0	0	0				
P_5	0	0	0	1	0	0	0	0				

b) **[Partial Marking, theme of algo should be correct]**

1. Let *Work* and *Finish* be vectors of length m and n, respectively. Initialize *Work* = *Available*. For $i = 0, 1, ..., n-1$, if $Allocation_i \neq 0$, then $Finish[i]$ = *false*. Otherwise, $Finish[i]$ = *true*.
2. Find an index i such that both
   a. $Finish[i] == false$
   b. $Request_i \leq Work$

   If no such i exists, go to step 4.
3. $Work = Work + Allocation_i$
   $Finish[i] = true$
   Go to step 2.
4. If $Finish[i] == false$ for some i, $0 \leq i < n$, then the system is in a deadlocked state. Moreover, if $Finish[i] == false$, then process P_i is deadlocked.

c) **[2+1 Marks with all calculations and No Deadlock]**

Work and availability are the same in the following calculation.

Because P_4's $Request = [1,0,0,0] \leq Available = [2,0,0,0]$, P_4 runs and returns is $Allocation = [0,1,0,1]$ making the new $Available = [2,0,0,0] + [0,1,0,1] = [2,1,0,1]$

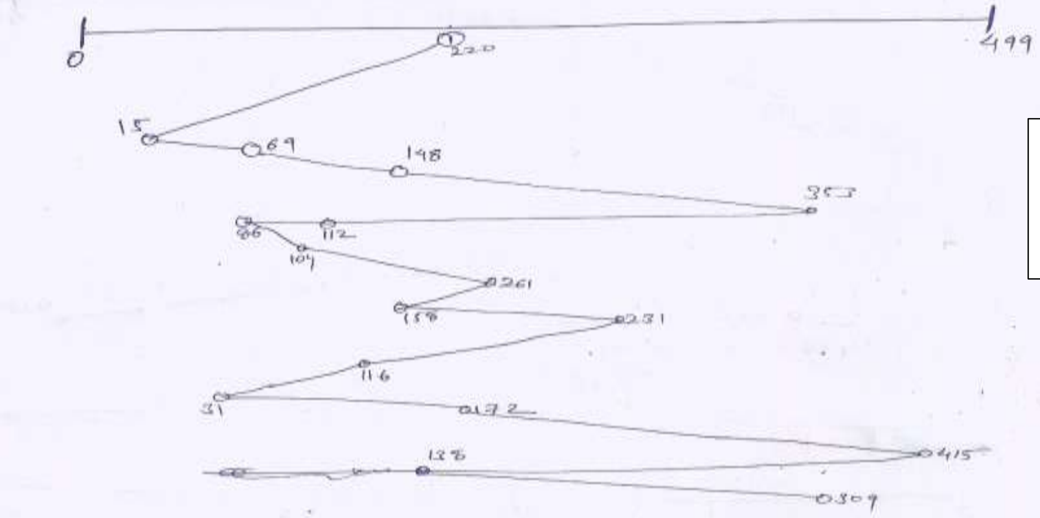
Then, we can run P_1 because P_1's $Request = [0,1,0,0] \leq Available = [2,1,0,1]$. After reclaiming P_1's $Allocation = [1,0,0,0]$, the new *Available* is old $Avaliable = [2,1,0,1] + P_1'$s $Allocation = [1,0,0,0] = [3,1,0,1]$.

The next process is P_3 because P_3's $Request = [0,0,0,1] \leq Available = [3,1,0,1]$. After P_3 finishes its work, its $Allocation = [0,0,1,0]$ is returned to $Available = [3,1,0,1] + [0,0,1,0] = [3,1,1,1]$:

Now we can run P_2 and the yields new available is [3 1 1 1]+[0 1 0 0]= [3 2 1 1]

	Finally, we can run P_5 and all processes are done! And new Available is **[3 2 1 2]** **No Deadlock in the System**
5.	a) Explain the following security breaches in detail. [**.5 Mark Each, should cover the core concept**] • Trojan Horse • Trap Door, • Logic Bomb • Stack and Buffer Overflow. b) Differentiate between compiler and interpreter. (**at least four**) [**2 Mark (.5 marks for each right difference)**] c) Explain the different free space management techniques. Discuss the relevant merit and demerit of each one. [**3 Mark (1 mark per Techniques), at least 3 Techniques should be explained with proper merit and demerit**] **[CO-5][7 MARKS]**
SOL	c)**Free space management techniques:** -Bit Vector Techniques -Linked List -Grouping -Counting
6.	Consider the following disk access requests made by multiple processes in the given order. **15, 69, 148, 353, 112, 86, 104, 261, 158, 231, 116, 31, 172, 415, 138, 309** The disk has **500** cylinders (0 to 499), with the head currently positioned at cylinder 220 and moving towards the disk's near end. The average power dissipation when moving the head over 50 cylinders is 7 milliwatts, and 2 milliwatts when reversing the direction of the head movement once. The power dissipation caused by rotational latency and switching of the head between platters is negligible. a) Which disk scheduling algorithm (**FCFS, SCAN, SSTF, LOOK**) will produce the second shortest average seek time? Display the head movements and compute the average seek time for each algorithm. **[4 MARKS]** [**1 Mark for each** Algo. **Diagram =.5 Marks** and **Correct Seek Time =.5 Marks**] b) Using the **Shortest Seek Time First** and **C-Look** disk scheduling algorithm computes total power consumption in milliwatts to satisfy all of the above disc requests. **[3 MARKS]** **[1.5 MARKS Each, Calculation and correct output should be there]** **[CO-5][7 MARKS]**

FCFS



.5 mark for diagram

.5 mark for total head movement

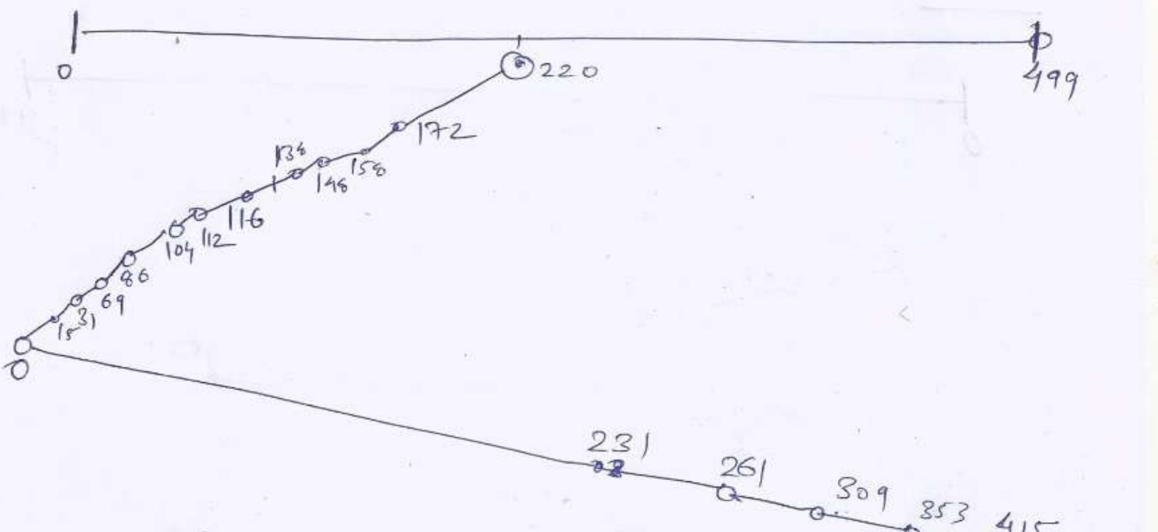
220 − 15 = 205
353 − 15 = 338
353 − 86 = 267
261 − 86 = 175
261 − 158 = 103
231 − 158 = 73
231 − 31 = 200
415 − 31 = 384
415 − 138 = 277
309 − 138 = 171

2193

SCAN

.5 mark for diagram

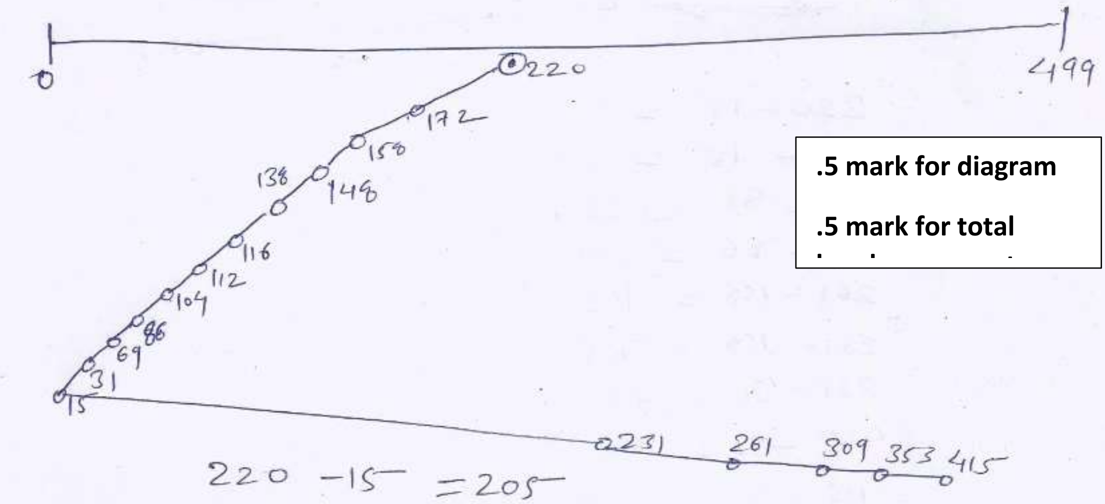
.5 mark for total head movement



220 − 0 = 220
415 − 0 = 415

635

~~SSTF~~ LOOK



.5 mark for diagram

.5 mark for total

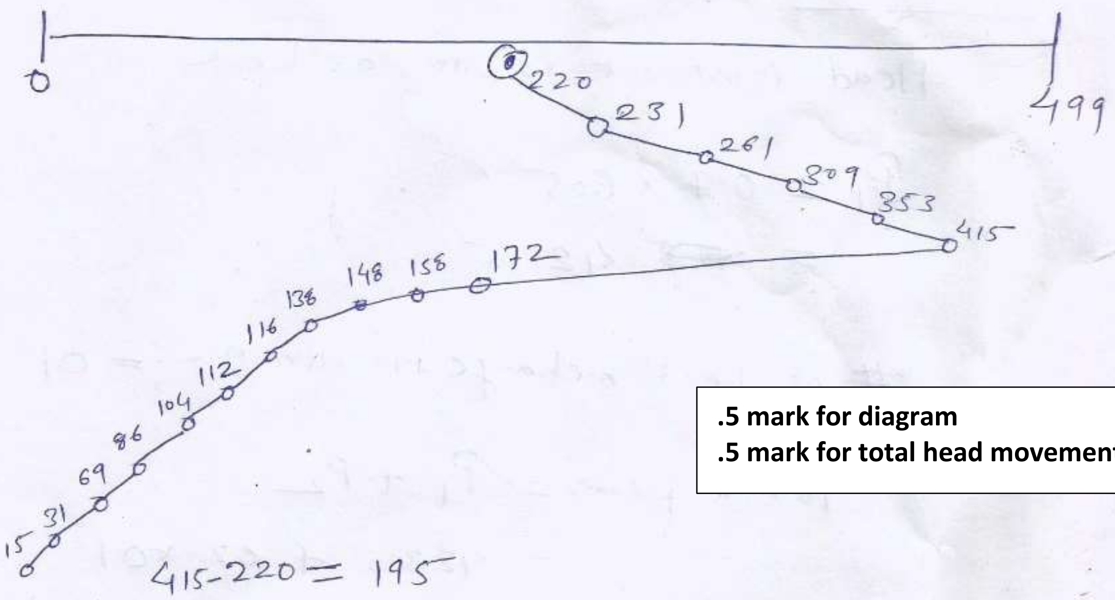
$220 - 15 = 205$
$415 - 15 = 400$

605

Second highest

(b)

SSTF

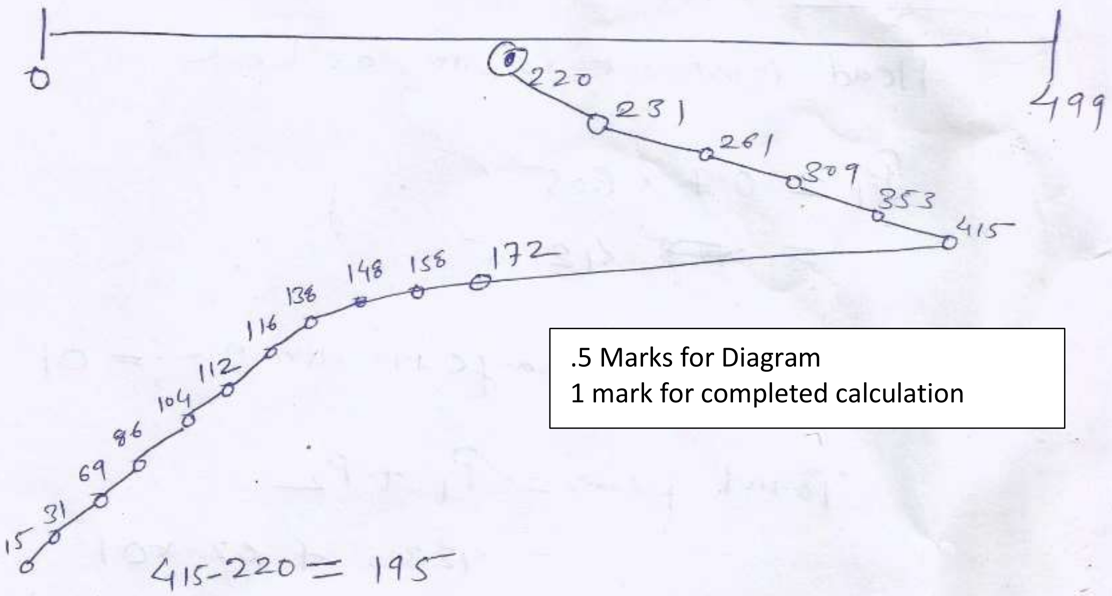


.5 mark for diagram
.5 mark for total head movement

$415 - 220 = 195$
$415 - 15 = 400$

595

(b) SSTF



.5 Marks for Diagram
1 mark for completed calculation

$415 - 220 = 195$
$415 - 15 = 400$
595

power dissipated by 50 movements. $P_1 = 0.7 \times 595$
$= 416.5$ mW

power dissipated in reversing head direction once $= 2$ mW

Number of time head changes its direction $=$ 01

Total $= P_1 + P_2$

$=$ = 416.5+2*1 = 420.05 mw

$\boxed{= 420.5 \text{ mW}}$

(b)

C-Look

.5 marks for diagram.

1 mark for complete calculation

Head movement is same as Look

$P_1 = 0.7 \times 605$

$= \cancel{423}\ 423.5$

# of head change its direction. = 01

Total power = $P_1 + P_2$

$= 423.5 + 02 \times 01$

$= \boxed{425.5\ mW}$