Programme Name: ______________**BCS**__________________________

Course Code**:** ___**CSC 1016**_______

Course Name: ____________________**Operating System**_______________________________

Open Book Examination

Date of Submission: ___________10/6/2020___________________

Submitted By:

Student Name: **CSC 1016_Dipesh Tha Shrestha**

IUKL ID: **041902900028**

Semester: **Second Semester**

Intake: **September 2019**

Submitted To:

Faculty Name: **Kushal Regmi**

Department: **LMS**

1. Differentiate between the following terms:

a. Authentication and Authorization

Authentication	**Authorization**
Authentication confirms your identity to grant access to the system.	Authorization determines whether you are authorized to access the resources.
Authentication usually requires a username and a password.	Authentication factors required for authorization may vary, depending on the security level.
For example, students of a particular university are required to authenticate themselves before accessing the student link of the university's official website. This is called authentication.	For example, authorization determines exactly what information the students are authorized to access on the university website after successful authentication.

b. Virus and Worms

Virus	**Worms**
The virus needs human help to execute and spread.	Worms automatically execute and spread.
Virus attaches itself with the host and spread where the host reaches.	Worms don't need a host and exploit the vulnerability of a network to spread.
Viruses destroy, damage, or alter the files in the infected computer.	Worms don't affect the file but increase the resource usage to crash the system or network.

Virus spreading speed is low compared to worms.	Worms spreading speed is fast, and it quickly infects multiple computers or networks.

2. One of the design decisions in operating system memory management is the choice between swapping and paging. Define their respective roles in operating system memory management

Answer:

Memory Management is the process of controlling and coordinating computer memory, assigning portions known as blocks to various running programs to optimize the overall performance of the system.

The respective role of swapping is to offering direct access to the memory images and role of paging is to allows OS to retrieve processes from the secondary storage into the main memory in the form of pages in operating system memory management.

3. System Calls provide an interface to the services provided by the OS.

a. List out the types of System Calls.

Types of system calls are

Process Control
File Management
Device Management
Information Maintenance
Communication

b. How the different Device Management System Calls are used in OS. Explain them.

The interface between a process and an operating system is provided by system calls. In general, system calls are available as assembly language instructions. They are also included in the manuals used by the assembly level programmers. System calls are usually made when a process in user mode requires access to a resource. Then it requests the kernel to provide the resource via a system call. System call offers the services of the operating system to the user programs via API (Application Programming Interface). System calls are the only entry points for the kernel system. For example, if we need to write a program code to read data from one file, copy that data into another file. The first information that the program requires is the name of the two files, the input and output files.

4. With the definition of Critical Section, explain the Solution to Critical Section Problem.

The critical section is a code segment where the shared variables can be accessed. An atomic action is required in a critical section i.e. only one process can execute in its critical section at a time. All the other processes have to wait to execute in their critical sections.

Solution to the Critical Section Problem

The critical section problem needs a solution to synchronize the different processes. The solution to the critical section problem must satisfy the following conditions −

- **Mutual Exclusion**

  Mutual exclusion implies that only one process can be inside the critical section at any time. If any other processes require the critical section, they must wait until it is free.

- **Progress**

Progress means that if a process is not using the critical section, then it should not stop any other process from accessing it. In other words, any process can enter a critical section if it is free.

- **Bounded Waiting**

  Bounded waiting means that each process must have a limited waiting time. It should not wait endlessly to access the critical section.

SECTION B

1. Assume that FIVE (5) processors enter ready state in a sequence as below:

Process	Arrival Time	Burst Time
P1	0	4
P2	1	7
P3	2	14
P4	3	5
P5	4	2

   a. Draw Gantt chart if the processor schedulers use the following scheduling algorithms:

i. FCFS.

CSC 1016 Dipesh Tha Shrestha.

Process	Arrival Time	Burst Time	Completion Time	TAT
P1	0	4	4	4
P2	1	7	11	10
P3	2	14	25	23
P4	3	5	30	27
P5	4	2	32	28

(i) FCFS

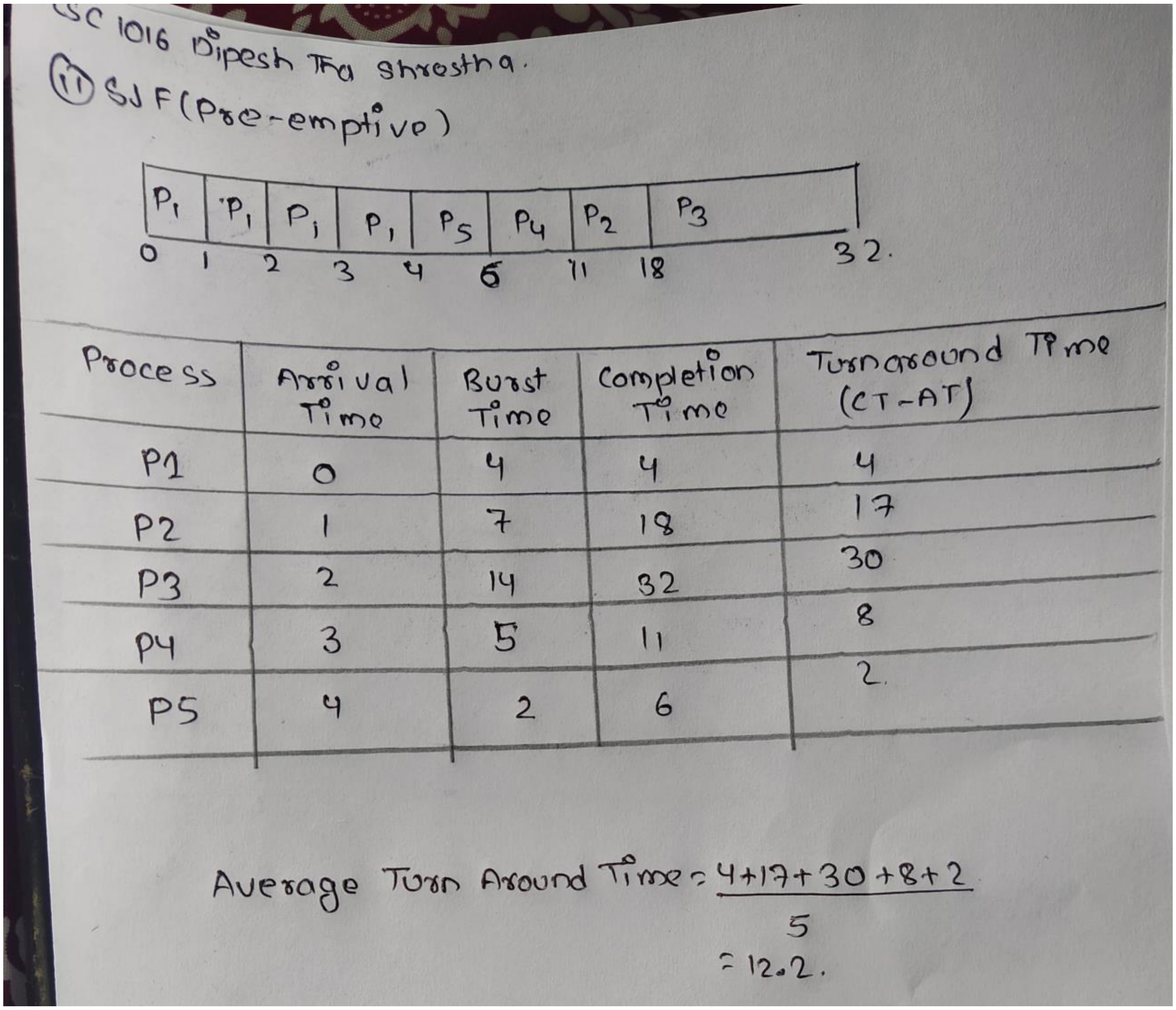
P1	P2	P3	P4	P5

0 4 11 25 30 32

$$\text{Average TAT} = \frac{4+10+23+27+28}{5}$$

$$= 18.4$$

ii. SJF (Pre-emptive).



CSC 1016 Dipesh Tha shrestha.

(ii) SJF (Pre-emptive)

P1	P1	P1	P1	P5	P4	P2	P3

0 1 2 3 4 6 11 18 32.

Process	Arrival Time	Burst Time	Completion Time	Turnaround Time (CT-AT)
P1	0	4	4	4
P2	1	7	18	17
P3	2	14	32	30
P4	3	5	11	8
P5	4	2	6	2.

$$\text{Average Turn Around Time} = \frac{4+17+30+8+2}{5} = 12.2.$$

iii. Round Robin with Time Quantum equal to 4 milliseconds

CSC 1016 Dipesh Tha Shrestha

(iii) Round Robin with Time Quantum equal 4 milliscand

P1	P2	P3	P4	P5	P2	P3	P4	P3	P3
4	8	12	16	18	21	25	26	30	32

Process	Arrival Time	Burst Time	Completion Time	Turnaround Time.
P1	0	4	4	4
P2	1	7	21	20
P3	2	14	32	30
P4	3	5	26	23
P5	4	2	18	14

$$\text{Average Turn Around Time} = \frac{4+20+30+23+14}{5}$$

$$= \frac{91}{5}$$

$$= 18.2.$$

b. Compare the average turnaround time for all the algorithms above

Therefore, The average turnaround time for all the algorithms are given below:

And the process of finding them in given in above diagram.

FCFS	18.4
SJF (Pre-emptive)	12.2
Round Robin with Time Quantum equal to 4 milliseconds	18.2

c. Based on your results of the average turnaround time in II, which algorithm is the best? Justify your answer.

Based on the result the Best Algorithm is SJF (Pre-emptive) Because it takes less time for the process than other algorithm

2. Consider the following snapshot of a system

	Allocation			Max			Available		
	U	V	W	U	V	W	U	V	W
A	0	0	1	0	0	1	1	5	2
B	1	0	0	1	7	5			
C	1	3	5	2	3	5			
D	0	6	3	0	6	5			
E	0	0	1	0	6	5			

Answer the following questions using the banker's algorithm:

a. What is the content of the matrix Need?

b. Is the system in a safe state?

Answer

CSC 1016_Dipesh Tha Shrestha

CSC 1016 Dipesh Tha Shrestha.

QNo2.

Need matrix is calculated by subtracting Allocating matrix from the max matrix.

	Need(Max Allocation).		
	U	V	W
A	0	0	0
B	0	7	5
C	1	0	0
D	0	0	2
E	0	6	4

Step 1: A need 0,0,0 and available 0,0,1
Need<= available à it is executed.
Work=work + allocation--à1,5,2+0,0,1
=1,5,3(available) a finish[0]= true

Step 2 : B needs 0,7,5 and available 1,5,3
Need<= available à it is not executed.
B-finish[1]=false

Step 3: C needs 1,0,0 and available 1,5,3
Need<= available à it is executed.
Work=work + allocation--à1,5,3+1,3,5
=2,8,8 (available) a finish[2]= true

Step 4 : D needs 0,0,2 and available 2,8,8
Need<= available à it is executed.
Work=work + allocation--à2,8,8+0,6,3
2,14,11 (available) a finish[3]= true

Step 4 : E needs 0,6,4 and available 2,14,11
Need<= available à it is executed.
Work=work + allocation--à2,14,11+0,0,1
2,14,12 (available) a finish[4]= true

Step 5 : B needs 0,7,5 and available 2,14,12
Need<= available à it is executed.
Work=work + allocation--à2,14,1+0,0,1
2,14,12 (available) a finish[5]= true
Allocation-à U=2, V=9, W=10
Total=available + work
=1,5,2+2,9,10
=3,14,12

WE CLAIM THAT THE SYSTEM IS IN SAFE STATE
<A, C, D, E, B>-àSafe sequence

3. Memory Management is the process of controlling and coordinating computer memory and assigning portions. Here is given five memory partitions of 100K, 500K, 200K, and 300K (in order).

CSC 1016_Dipesh Tha Shrestha

a. How would the first-fit, best-fit, and worst-fit algorithms place processes of 190 K, 288 K, 112 K, and 250 K (in order)?

CSC 1016 Dipesh Tha Shrestha.

Q No 3

Given processes are 190k, 288k, 112k, 250k

Given postition.

→

100 k	500k	200k	300k

After first fit algorithm, process are allocated as follows

100 k	190k	288k	22k	112k	88k	250 k	50 k

Fig: First fit.

In Best-fit algorithm, process are allocated as follows:-

100 k	112K	250k	138 k	190k	10K	288k	12 k

fig: Best fit.

In worst-fit algorithm, process are allocated as follows-

100 k	190k	288 k	22k	200 k	112k	188k

fig- worst fit.

c. Which algorithm makes the most efficient use of memory?
   Best fit Algorithm makes the most efficient use of memory.

4. Consider the traffic deadlock depicted in Figure.

a. Show that the FOUR (4) necessary conditions for deadlock hold in this example.

Ans: The four necessary conditions for deadlock hold in the given example are as follows:
i)Mutual exclusion The resources involved must be unshareable; otherwise, the processes would not be prevented from using the resource when necessary. At least one resource must be held in a non-sharable mode; If any other process requests this resource, then that process must wait for the resource to be released.

ii)Hold and wait or partial allocation The processes must hold the resources they have already been allocated while waiting for other (requested) resources. If the process had to release its resources when a new resource or resources were requested, deadlock could not occur because the process would not prevent others from using resources that it controlled.

iii)No pre-emption The processes must not have resources taken away while that resource is being used. Otherwise, deadlock could not occur since the operating system could simply take enough resources from running processes to enable any process to finish.

iv)Resource waiting or circular wait A circular chain of processes, with each process holding resources which are currently being requested by the next process in the chain, cannot exist. If it does, the cycle theorem (which states that "a cycle in the resource graph is necessary for deadlock to occur") indicates that deadlock would occur. A set of processes { P0, P1, P2, . . ., PN } must exist such that every P[i] is waiting for P[(i + 1) % (N + 1)].

b. State THREE (3) ways to handle deadlock problem in this system.

Ans: The three ways to handle deadlocks are as follows:

i)Deadlock prevention or avoidance - Do not allow the system to get into a deadlocked state. It is important to prevent a deadlock before it can occur. So, the system checks each transaction before it is executed to make sure it does not lead to deadlock. If there is even a slight possibility that a transaction may lead to deadlock, it is never allowed to execute.

ii)Deadlock detection and recovery -Deadlock can be detected by the resource scheduler as it keeps track of all the resources that are allocated to different processes. After a deadlock is detected, it can be handed using the given methods − All the processes that are involved in the deadlock are terminated. This approach is not that useful as all the progress made by the processes is destroyed. Resources can be preempted from some processes and given to others until the deadlock situation is resolved.

iii)Ignore the problem all together - If deadlocks only occur once a year or so, it may be better to simply let them happen and reboot as necessary than to incur the constant overhead and system performance penalties associated with deadlock prevention or detection. This is the approach that both Windows and UNIX take.