Operating System Solution Spring Semester -2020

10th JUNE 2020 & 11 PM to 1 PM For IT & CSSE Branch Question of SLOT – 2 (Solution to be done by)	
Section A Question No. 1 (A & B)	Prof. Bindu Agarwalla
Section A Question No. 2 (A & B)	Prof. Tanmoy Maitra
Section B Question No. 3 (A & B)	Prof. (Dr.) Alok Kumar Jagadev
Section B Question No. 4 (A & B)	Prof. Subhasis Dash
Section B Question No. 5 (A & B)	Prof. (Dr.) Biswajit Sahoo

SECTION – A

1.(A) Justify which scheduling algorithm, as an operating systems designer, would you prefer to implement? [2.5 Marks]

Slot 2 (IT & CSSE)

1. a> Ans: Marks will be awarded for the appropriate justification given for the chosen scheduling algorithm.

1.(B) Consider process arrival as given below where m = 10 * (1+ (right most significant digit of your Roll No % 9)). (ex:- for Roll No. 18056**3**, m=10 *(1+(**3**%9)) = 40):

Process	CPU Burst Time (ms)	Arrival Time
A	40	0
B	55	10
C	60	m
D	m	30
E	10	50

Calculate the following for *round robin* (time quantum = 20 ms) CPU scheduling algorithm

i. Average waiting time

ii. Turnaround time for each process

iii. Order of completion

iv. Minimum time quantum so that *round robin* will act as *first come first serve* scheduling.

ANS:-

1. b) Ans:

Process	CPU BT (ms)	AT	CT	(ii) TAT	(i) WT
A	~~40~~ ~~20~~ 0	0	60	60	20
B	~~55~~ ~~35~~ 15	10	185	175	120
C	~~60~~ ~~40~~ 20	m(40)	205	165	165
D	m(~~40~~) ~~20~~	30	150	120	80
E	~~10~~ 0	50	130	80	70
					79ms

m = 10 * (1 + right most significant digit of your Roll No % 9))

For, Roll No. = 180563 TQ = 20ms.

m = 10 * (1 + 3%9)

= 40

~~ABACB~~ ~~ABABCBEDB~~ B C

A	B	A	D	C	B	E	D	C	B	C
0–20	20–40	40–60	60–80	80–100	100–120	120–130	130–150	150–170	170–185	185–205

iii) Order of completion:

A → E → D → B → C

i) Avg. WT = 79ms #

iv) Minimum time quantum so that round robin will act as first come first serve scheduling.

max (BT) = 60ms. #

Calculation of value of m:-

where m = 10 * (1+ (right most significant digit of your Roll No % 9)). (ex:- for Roll No. 18056**3**, m=10 *(1+(**3**%9)) = 40):

Roll No. last digit as value of "m"	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
CPU burst time and Arrival time represented as "m" calculated as "10*(1+(**m**%9))**"**	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**10**

2. (A) How can one have concurrent execution of threads/processes without having parallelism? [2.5 Marks]

ANS:- Explain with Justification [2.5 Marks]

2. (B) During a lockdown period, you are playing Ludo with your *n* number of family members where, n < 4. Assume that all the tokens are in the starting square and consider following scenarios:

If the outcome of the dice is 6 (six), then you can move your one token and get the chance to play dice again; otherwise, you can move your one token and give the dice to the next player. Think of the players as processes which should be synchronized.

Write a code for it using semaphore. Your answer should first (i) list what synchronization and/or what critical section problems you have to solve, (ii) define what semaphore(s) you have to use (including their initial value(s)), and (iii) then write the pseudo code for it. You can assume that WAIT( ) and SIGNAL( ) are primitive calls on a semaphore with their usual meanings.

[10 Marks]

ANS:-

Shared resource is dice.

i) a. if the outcome of the dice is 6, then all the other players must wait.

b. if the outcome of the dice is other than 6, then immediate next player gets access of the dice and other players must wait.

ii) Counting semaphores p[0] = 1, p[1]=…=p[n-1]=0 ensure that which process will get access of dice next.

iii)
```
while ( True )
{
   Wait ( p[ i ] )
        While ( True )
           {
             If ( Outcome = = 6 )
               {
                  // Move the Token
                   Continue;
                }
             else
                 {
                  // Move the Token
                    Break;
                 }
             }
     Signal ( p [ ( i + 1 ) % n ] )
   }
```

SECTION - B

3. (A) Consider a system consisting of four resources of the same type that are shared by three processes, each of which needs at most two resources. Show that the system is deadlock-free. [4.5 Marks]

ANS:-

Suppose the system is deadlocked. This implies that each process is holding one resource and is waiting for one more. Since there are three processes and four resources, one process must be able to obtain two resources. This process requires no more resources and, therefore it will return its resources when done.

3. (B) Consider the following snapshot of a system with 5 processes (P_0, P_1, P_2, P_3, P_4) and 3 resource types (R_0, R_1, R_2). [8 Marks]

Process	Max Need			Allocation			Available		
	R_0	R_1	R_2	R_0	R_1	R_2	R_0	R_1	R_2
P_0	4	0	1	0	0	1	1	T	2
P_1	1	7	U	1	0	0			
P_2	2	3	5	1	3	5			
P_3	0	2	5	0	6	3			
P_4	3	6	5	2	0	1			

Answer the following questions using the banker's algorithm:

(where U = N MOD 2 +1; T= N MOD 3 + 1; assume N= Your Roll Number)

i. Find the safe sequence if the system in a safe state?

ii. If a request from process P1 arrives for (0, 4, 2), can the request be granted immediately?

ANS:-

U = 1 or 2
T = 1 or 2 or 3

Here, it is considered U=1 and T=1
The matrix will be

Process	Max Need			Allocation			Available		
	R_0	R_1	R_2	R_0	R_1	R_2	R_0	R_1	R_2
P0	4	0	1	0	0	1	1	1	2
P1	1	7	1	1	0	0			
P2	2	3	5	1	3	5			
P3	0	2	5	0	6	3			
P4	3	6	5	2	0	1			

So, the need matrix will be

Process	Max Need			Allocation			Need			Available		
	R_0	R_1	R_2	R_0	R_1	R_2	R_0	R_1	R_2	R_0	R_1	R_2
P0	4	0	1	0	0	1	4	0	0	1	1	2
P1	1	7	1	1	0	0	0	7	1			

P2	2	3	5	1	3	5	1	0	0			
P3	0	2	5	0	6	3	0	-4	2			
P4	3	6	5	2	0	1	1	6	4			

i) Find the safe sequence if the system in a safe state?

Here, the data for process P3 is erroneous.
The sequence is not a safe sequence,
After execution of P2, the available resource set will become (2, 4, 7). If we will not consider P3, no other process among {P0, P1, P4} will able to execute within these available resources.

ii) If a request from process P1 arrives for (0, 4, 2), the value of available will become (0, -3, 0), which is impossible. So it will be an unsafe sequence.

Suggestion:	Marks
Computing the value of U and T	2 marks
Computing Need Matrix	1 mark
In (i): discussing about –ve value + considering Unsafe sequence	2.5 marks
In (ii): discussing how to compute the matrix for additional requirement of a process + considering Unsafe sequence	2.5 marks

4. (A) Show implementation of fixed and variable partition schemes in the main memory. Illustrate, how different types of memory fragmentation problem occurs due to these partition schemes during the allocation of processes in memory. Give examples from each to explain better. [4.5 Marks]

ANS:-

Fixed partition and variable partition → Explanation [2 Marks]

types of memory fragmentation problem → Internal and External Fragmentation Problem

Explanation with Example [2 .5 Marks]

4. (B) A multiprogramming system is available with a variable partition scheme. The system uses a set of free memory list to track different available memory spaces. The current list contains the entries of 270KB, 100KB, 450KB, 350KB, 175KB, and 600KB as free. The system receives requests for multiple processes like 585KB, 270KB, 346KB, 140KB, 435KB, and 90KB in order. Draw and explain the final allocation of the process by using the following dynamic memory allocation strategies like the best fit, first fit, and worst fit.

[8 Marks]

ANS:-

Process List → 585KB, 270KB, 346KB, 140KB, 435KB, and 90KB

First Fit (2.5 Mark)

Free Space Available	Process Allocated	Space Available after Allocation of Process
270KB	270KB	**0 KB**
100KB	90KB	**10 KB**
450KB	346KB	**104 KB**
350KB	140KB	**210 KB**
175KB		**175 KB**
600KB	585KB	**15 KB**
		Total Unused Space = 514 KB

Process with 435KB remains unallocated.

Best Fit (2.5 Mark)

Process List → 585KB, 270KB, 346KB, 140KB, 435KB, and 90KB

Free Space Available	Process Allocated	Space Available after Allocation of Process
270KB	270 KB	**0 KB**
100KB	90 KB	**10 KB**
450KB	435 KB	**15 KB**
350KB	346 KB	**04 KB**
175KB	140 KB	**35 KB**
600KB	585 KB	**15 KB**
		Total Unused Space = 79 KB

Worst Fit (3 Mark)

Process List → 585KB, 270KB, 346KB, 140KB, 435KB, and 90KB

Free Space Available	Process Allocated	Space Available after Allocation of Process
270KB	140 KB	**130 KB**
100KB		**100 KB**
450KB	270 KB	**180 KB**
350KB	346 KB	**04 KB**
175KB	90 KB	**85 KB**
600KB	585 KB	**15 KB**
		Total Unused Space = 514 KB

Process with 435KB remains unallocated.

5. (A) A manufacturer wishes to design a hard disk with a capacity of M GB. If the technology used to manufacture the disks allows 1024-byte sectors, 2048 sectors/track, and 4096 tracks/platter, how many platters are required? (Assume M=10*(1+ (right most digit of your Roll No % 9)); Ex:- for Roll No. 183456; M=10*(1+(6 % 9)) i.e. 70) [4.5 Marks]

ANS:- No of bytes in a platter = 4096 * 2048 *1024 B

$=2^{12} * 2^{11} * 2^{10}$ B

$=2^{3} * 2^{30}$ B

=8GB

For Roll No. 18345**6**; M=10*(1+(**6** % 9))

M= 70

Now, No. of platter required for 70GB disk = 70GB / 8GB

=8.75

So, Number of platters = 9 (Ans)

Generalized Answer

RMD of Roll No	0	1	2	3	4	5	7	8	9
Answer	2	3	4	5	7	8	10	12	2

5. (B) Compare and contrast chained allocation with indexed allocation technique of file allocation. What is an Inode and what role does it play? What corresponds to Inode in the MS environment? [8 Marks]

ANS:

Theory answer available in Text Book.

4 marks for first part

2 marks for second part

2 marks for third part