Operating System (CSC 330)

BSCS- 5th A

Bahria University Discovering Knowledge	**Department of Computer Science** **Bahria University, Lahore Campus**

Assignment No 3

Due Date: **Total marks = 5**

Name: **Roll No:**

CLOs
CLO2: Apply the concepts of memory management, I/O management CPU management and processor management etc.

Guidelines:

- Assignments can be in groups (max two members).
- Assignments should be clear, concise, and to the point.
- Sharing assignment ideas, copying the content, and solution is permissible with your class fellows (if anyone found with such case, maximum marks of any member will be divided by number of persons who copied with each other).
- Please submit the assignment as per the given deadline. Late submitters will be penalized.
- Submit the hard copy during class (on the day of deadline) and scanned copy at LMS on or before due date.

Marks Evaluation Criteria:

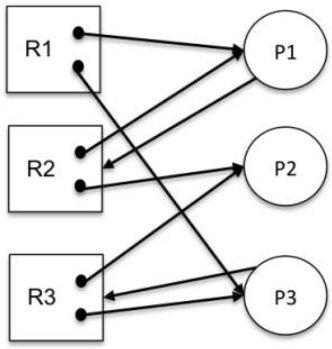
Feature	Assessment Criteria	Marks
Present ability	The assignment is clean, properly formatted, table of contents, title page contains the name/logo of the institute, submitter's name, teacher's name.	0.5
Assignment Content	Questions solved correctly and presented effectively.	4
Deadline Met	Assignments are submitted within a given deadline.	0.5
Deadline Missed	Assignment deadline is missed / and submitted late.	-0.5 / and -0.5 per day

Important Note:

- All the below questions are sample questions with their possible solutions.
- You are now required to design your own question with the same pace but with different stats in each question.
- Matching questions would be divided by the number of groups who tried to cheat.

Question – 1:

Consider the following resource allocation graph, and answer the below-mentioned questions:



a) Does the above allocation graph contain a deadlock? Show the steps/procedure that you took to reach your conclusion, a simple YES/NO answer is not sufficient.
Solution: No. A possible execution sequence is P2, P1, P3.

b) Assume now that P2 also demands resource R1. Will this demand put the system into a deadlock? Show the steps/procedure that you took to reach your conclusion, a simple YES/NO answer is not sufficient.
Solution: Yes, if P2 demands R1, none of P1, P2, P3 will get all resources they need not to be processed. Therefore, there is a deadlock.

c) Add to the original allocation graph an additional process P4 that demands an instance of R1. Does the allocation graph contain a deadlock? Show the steps/procedure that you took to reach your conclusion, a simple YES/NO answer is not sufficient.
Solution: No. The only possible execution sequence is P1, P2, P3.

Question – 2:

Consider a system with five processes, P0, P1, P2, P3, and P4, and three resource types, A, B, and C. The current state of the system is represented in the table below:

Processes	Maximum Need			Allocation			Available		
	A	B	C	A	B	C	A	B	C
P0	4	3	3	1	1	2	2	1	0
P1	3	2	2	2	1	2			
P2	9	0	2	4	0	1			
P3	7	5	3	0	2	0			
P4	1	1	2	1	1	2			

a) Using the banker's algorithm, calculate the values of the need and available matrices. Determine whether the system is currently in a safe state or an unsafe state.
Solution:

Processes	Maximum			Allocation			Need			Available		
	A	B	C	A	B	C	A	B	C	A	B	C
P0	4	3	3	1	1	2	3	2	1			
P1	3	2	2	2	1	2	1	1	0			
P2	9	0	2	4	0	1	5	0	1			
P3	7	5	3	0	2	0	7	3	3			
P4	1	1	2	1	1	2	0	0	0			

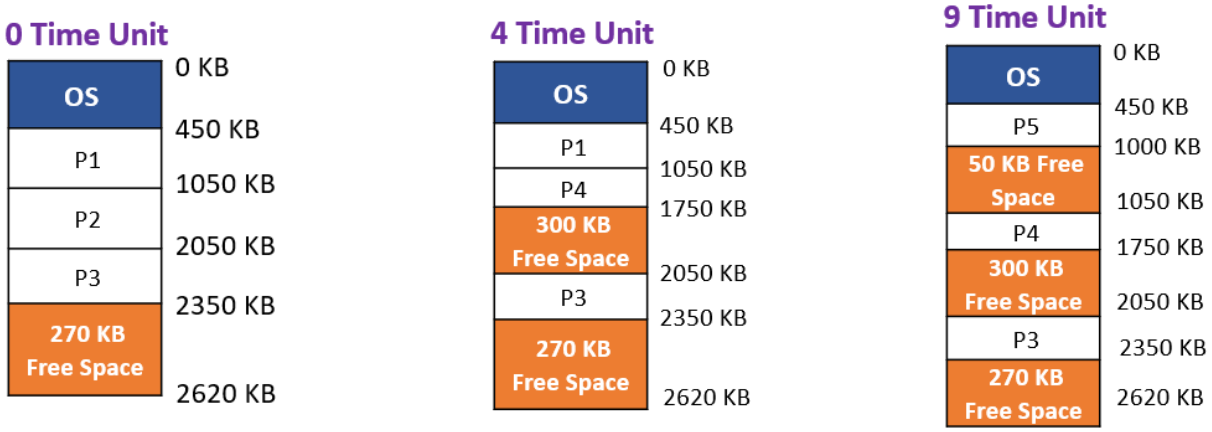
Safe Sequence: < P1, P4, P0, P2, P3> You are required to show the stats of work and finish array as well.

b) Suppose Process P2 requests (1, 0, 1) resources. Will the request be granted immediately? Explain your answer, a simple YES/NO answer is not sufficient.
Solution: Yes, resources will be granted as available resources (2, 1, 0) are greater than requested (1, 0, 1).

Question – 3:

Consider a memory partition of size 2620KB in which the first 0KB to 450KB memory is allocated to Operating System. Rest, there are 5 processes requiring memory sizes 600KB, 1000KB, 300KB, 700KB, and 550KB, and the execution time of each process is as 9, 4, 19, 7, 14 in order. Apply the suitable memory allocation scheme to allocate memory to each process.

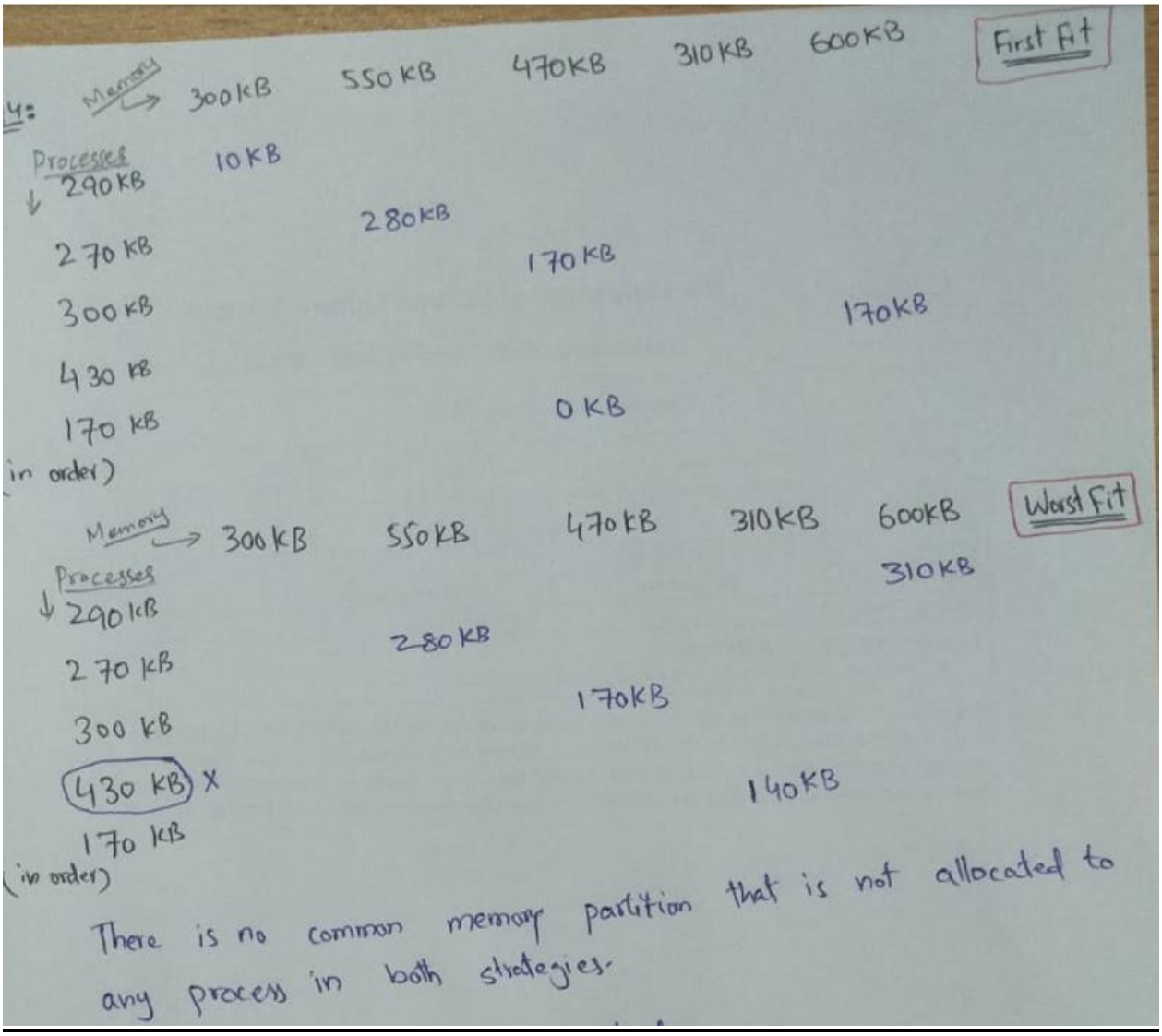
Solution:



Question – 4:

Suppose there are five memory partitions of fixed sizes as 300KB, 550KB, 470KB, 310KB, 600KB, and five processes that require memory as 290KB, 270KB, 300KB, 430KB, and 170KB (in order). Allocate the memory to each process from the given memory partitions applying First Fit and Worst Fit memory allocation strategies. Also describe that is there any common memory partition(s) which is(are) not allocated to any process in both strategies.

Solution:



Question – 5:

You are supposed to implement variable partition memory allocation schemes such as First Fit, Best Fit, and Worst Fit. You can implement it in C, C++, java, or Python.

Note: It can be implemented on the system and soft-form code may be added to submit the assignment.

Sample Input Output System:

First Fit:

```
Please Enter number of Blocks: 4

Please Enter size of all blocks: 20
30
50
70

Please Enter number of Processes: 3

Please Enter Memory Required for each process:10
20
60

Process_no      Process_size    Block_no        Block_size      Fragment
1               10              1               20              10
2               20              2               30              10
3               60              4               70              10
Process returned 0 (0x0)   execution time : 27.226 s
Press any key to continue.
```

Best Fit:

```
Please Enter number of Blocks: 4

Please Enter size of all blocks: 30
20
70
100

Please Enter number of Processes: 3

Please Enter Memory Required for each process:10
70
20

Process_no      Process_size    Block_no        Block_size      Fragment
1               10              2               20              10
2               70              3               70              0
3               20              1               30              10
Process returned 0 (0x0)   execution time : 24.649 s
Press any key to continue.
```

Worst Fit:

```
Please Enter number of Blocks: 4

Please Enter size of all blocks: 20
30
100
40

Please Enter number of Processes: 4

Please Enter Memory Required for each process:10
30
20
10

Process_no      Process_size    Block_no        Block_size      Fragment
1               10              3               100             90
2               30              4               40              10
3               20              2               30              10
4               10              1               20              10
Process returned 0 (0x0)   execution time : 36.983 s
Press any key to continue.
```

Do your own, some One is watching!

Best of Luck!