8/8 Questions Answered

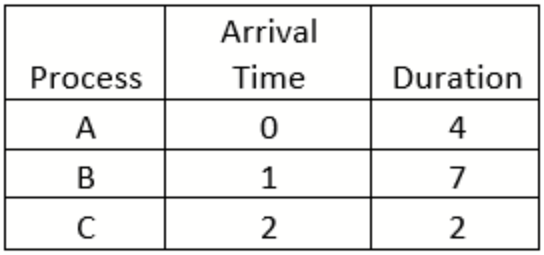
Saved at 6:28 PM

Check-in Quiz05, Scheduling & File System

Q1 Scheduling

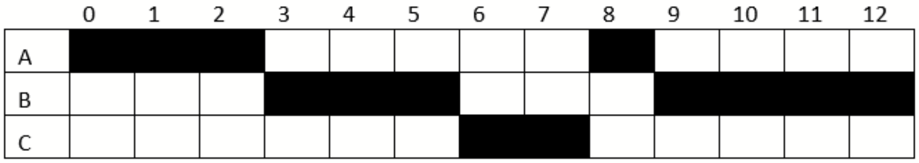
5 Points

Consider the following processes that the scheduler sees:



Process	Arrival Time	Duration
A	0	4
B	1	7
C	2	2

If we were to schedule these with a time quantum of 3 we would get the following execution diagram:



	0	1	2	3	4	5	6	7	8	9	10	11	12
A	■	■	■						■				
B				■	■	■				■	■	■	■
C							■	■					

If we count the number of context switches, we get 4 of them (not counting the loading of the first process and the unloading of the last process)

In this scenario, we made the following assumptions:

- All processes do not block for I/O or any resource.
- Context switching and running the Scheduler are instantaneous.
- If a process arrives at the same time as the running process' time slice finishes, the one that just arrived goes into the ready queue before the one that just finished its time slice.
- If a process would not finish before the time quantum is up, and there are no other processes to schedule, then the process runs for another time quantum without context switching.

Consider the case where out time quantum is shrunk from 3 time units to 2 time units. For each of the following, please answer what the new value would be and answer it with a number only:

Q1.1
1 Point

The finishing time for process A?

Hint: with a time quantum of 3, process A had a finishing time of 9

8

Explanation

Correct! A would get to run first for two time units but would have to wait its turn before it can run again. B and C would both get to run for 2 units each before A runs again finishing its last two units of time needed.

✓ **Correct**

Save Answer Last saved on **Dec 05 at 6:27 PM**

Q1.2
1 Point

The finishing time for process B?

Hint: with a time quantum of 3, process B had a finishing time of 13

13

Explanation

Correct! B is such a long process that all other jobs will be able to finish before it finishes. B will slowly make progress on its work but get pre-empted after 2 time units.

✔ **Correct**

Save Answer Last saved on **Dec 05 at 6:27 PM**

Q1.3
1 Point

The finishing time for process C?

Hint: with a time quantum of 3, process C had a finishing time of 8

6

Explanation

Correct! A would get to run first for two time units before being pre-empted to let B run. B would then run for two units before we switch to running C. The time Quantum is still long enough for C to finish before it would get pre-empted, so it finishes at time 6.

✔ **Correct**

Save Answer Last saved on **Dec 05 at 6:27 PM**

Q1.4
2 Points

The number of context switches there would be between processes A, B and C?

4

Explanation

Correct! There are still 4 context switches, but we get a better finishing time for some of our processes :)

✔ **Correct**

Save Answer Last saved on **Dec 05 at 6:27 PM**

Q2 Number of Seeks

3 Points

One of the things about Hard Disk Drives is that it takes a long time to seek to a new position on disk, so one major factor when deciding a disk allocation scheme is how many seeks it takes to access any specific block of a file.

Consider we have the a file containing 4 blocks of data, how many disk seeks do we need to make to read the 3rd block of the file?

You can assume we already know the physical block number of the first block of the file.

Q2.1 Contigouos Allocation

1 Point

If the disk allocation scheme is contiguous allocation:

0

1

2

3

4

None of the above

Explanation

Correct! If we know the physical block number of the first block in the file, we can calculate the physical block number for the third block and just directly seek to it and read it

✔ **Correct**

Save Answer Last saved on **Dec 05 at 6:27 PM**

Q2.2 Linked List Allocation
1 Point

If the disk allocation scheme is linked list allocation:

0

1

2

3

4

None of the above

Explanation

Correct! If we only know the physical block number of the first block in the file, have to go to that block and read the "next" pointer that is stored with that block. With that pointer we know the physical block number of the second block, and can go read it to get the physical block number of the third block. Once we have this, we can finally read the third block of the file.

✔ Correct

Save Answer

Last saved on **Dec 05 at 6:27 PM**

Q2.3 FAT
1 Point

If we are using a FAT allocation and we have already loaded the FAT table into memory:

0

1

2

3

4

None of the above

Explanation

Correct! If the FAT is already loaded into memory, we can "follow the linked list" like we did for linked list allocation. Instead of having to perform a disk read to find the physical block number of the next block, that information is stored in memory. This means we only need to read the block we are looking for once we have calculated its physical block number from the in-memory FAT.

✔ **Correct**

Save Answer Last saved on **Dec 05 at 6:27 PM**

Q3 FAT

2 Points

Consider the following FAT table:

index	value
0	
1	-1
2	5
3	9
4	
5	8
6	7
7	2
8	-1
9	1

If we say that file "ngnm.txt" starts at physical block 3, what are all of the blocks that make up that file?

Please answer as a list of comma separated numbers with no white space. Please list the blocks so that they are in the order they logically would be to compose the file.
For example: `7,3,12,5,4`

3,9,1

Explanation

Correct! We start at 3 and then stop at 1, not including -1 since that marks the end of the file

✔ Correct

Save Answer Last saved on **Dec 05 at 6:28 PM**

Save All Answers

Submit & View Submission ❯