**Lab 9**

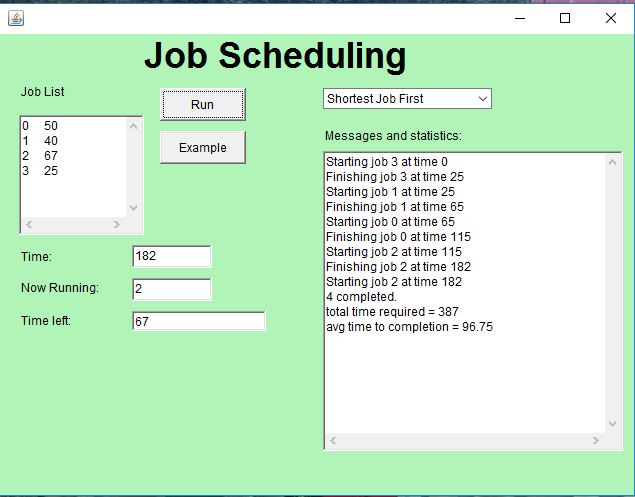
1. Lab 13 Operating Systems (Lab13\_Manual.pdf)

a. Exercise 2

1) Start the job scheduling app and click on Example button.

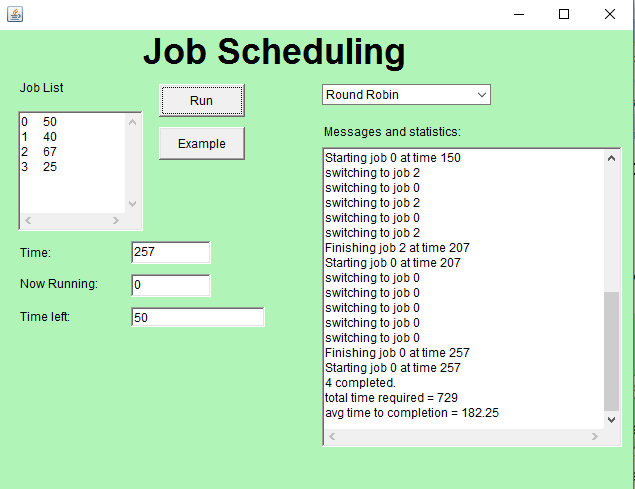
2) Run the program using first-come, first-served algorithm.

3) Select the Shortest Job first from the menu and click on run button. Does this method have the same total time requirement as FCFS algorithm? Note the average time of completion. Is it better? Take a screenshot.

ANS: It does not have same time as first-come, first-served algorithm ( first-come, first-served- 479 and shortest job first- 387). The average time of completion for First-come, first-served is 119.75 while that for Shortest Job first algorithm is 96.75. It is actually better than first-come, first- served because both total time and average time is lesser in Shortest job first.

4) Select Round Robin from the menu and click on the run button. Write down the average time to completion. How does it compare to the other two methods? Take a screenshot.

ANS: Average time to completion for Round Robin is 182.25. The average time as well as total time for round-robin is greater than the other two algorithms.



5) Looking at the log of activity, what is fundamentally different about round robin from the other two job scheduling algorithms? What kind of computing system would likely prefer to use round robin?

ANS: The fundamental difference about round robin is that we can set the quantum size.

2. Lab 14 Disk Scheduling (Lab14\_Manual.pdf)

a. Exercise 1

1) Start the disk scheduling app.

2) Click on example button.

3) Select FCFS from menu and click the run button. Time the app and see how much time it takes.

ANS: It takes about 28 seconds.

4) Select shortest seek time first and run. Measure the time. Also write down the order of the tracks that are visited.

ANS: It takes about 14seconds (approx that was 13:40)and the order is as follows:

52, 47, 44, 61, 87, 96, 25, 20, 4

(50 cannot be seen anywhere in the list)

5) Repeat step4 with SCAN Disk.

ANS: It also takes about 13 seconds and the order for this is :

47, 44, 25, 20, 4, 52, 61, 87, 96

6) Which took the least time?

ANS: The SCAN Disk took the least time that was 14 seconds (approx). But the time for Shortest Seek Time was also similar. There was difference of milliseconds.

7) Based on your observations, state what kind of request list would make first-come, first-served take up a lot of time. In other words, what pattern would the numbers in the request list have to follow so that FCFS consumes a lot of time?

ANS: For first-come, first-served algorithm to consume a lot of time, the numbers entered should have a lot of difference between them because it goes from number that was entered first to the second then third and so on. So, larger the difference between the numbers entered, larger is the time taken.

b. Exercise 2

1. Create a request list of five track numbers that will cause all three algorithms to visit the same tracks in the same order.

ANS: The request list may look like:

60, 65, 70, 75, 80

1. If shortest seek time first starts with the disk head positioned at either 0 or 99, instead of 50, which algorithm would it resemble: FCFS or SCAN? Why?

ANS: It would resemble SCAN because SCAN goes from 50 to peak on one side to the other side peek like from 50 to 0 to 99 or from 50 to 99 to 0.

1. Based on your past lab experiences with stack and queue, is the request list of the disk scheduler a stack or queue?

ANS: If we talk about first-come, first-served algorithm then it is a queue but for the other two it is neither stack nor queue.

c. Exercise 3

1. Start the disk scheduling app and type the following into requests text area:

8, 20, 35, 80, 10, 90, 5, 87, 26, 94

1. Select FCFS and click on run. When disk heads has reached 35, type “30” into add area and press Enter. This will add a request to seek to track “30” to the list. What happens? Does the disk drive respond to new or not?

ANS: The disk drive runs as it was running earlier and consumes “30” at the end as it consumes those first which were entered first. It responds to the new but not at the same when it was entered.

1. Stop the app and remove “30”from the end of the list. Choose Shortest time first and rerun the app. When it consumes 35, type 30 into add area, and press Enter. Write down what happens.

ANS: It consumes “30” very fast after it is entered after “35” even it does not took time to represent 30 over there.

1. Stop the app, remove 30 from the end and choose SCAN. Run it, when it consumes 35, type “30” again. Write down what happens. Again, you must be fast.

ANS: It also does the same as done by Shortest time first algorithm that is it also consumed “30” right away after adding.

1. Which one is least responsive to new requests?

ANS: First-come, first-served was least responsive to the new requests because it consumes the numbers in the same order as those are entered.

1. Stop the app, remove 30 from the end of list, choose shortest seek time first and click on Run. Now try to “trap” the disk head into lower cluster by typing in disk tracks that are in the lower half of the disk drive, pressing Enter after each one. Write down what happens. You have to be quick and you may have to try it several times.

ANS: In this case the disk head traps at the lower half of the disk and never goes up.

1. Redo step6, but choose SCAN instead. Once again, try to “trap” the disk head into lower cluster by typing in disk tracks that are in lower half of the disk drive, pressing Enter after each one. Were you successful?

ANS: In this case, the head keeps in the lower cluster until we are adding the numbers in lower half but it goes up at last. So, I was unsuccessful to trap it .

1. In real life, disk drive may see a cluster of track requests that could trap it in one section of the disk drive. What implications does this have for programs that requested tracks outside the busy area?

ANS: It can have various implications like the earlier requests may never gets processed when the new keeps arriving that take priority being in one section or if the values are similar or near to each other. This is known as Starvation. They chose this term for the earlier terms that are being left behind or starved.

3. (Paged Memory Management) Given the following Page-Map Table (PMT), assume the page size and frame size are 1024:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Page | 0 | 1 | 2 | 3 | 4 |
| Frame | 4 | 5 | 8 | 1 | 3 |

a. What is the physical address associated with the logical address of <4,130>?

ANS: Logical address- <4,130>

Check PMT:

Page 4 maps to frame 3.

Physical address = frame\* frame size + offset

= 3\*1024 + 130 = 3072 + 130

Physical address = 3202

b. What is the logical address associated with the physical address of 5200?

ANS: Physical address = 5200 and page and frame size is 1024.

Frame number = address/ frame size = 5200/1024 = 5

Offset = remainder of the address over the frame size

Offset = 5200%1024 = 80

Check PMT:

Frame 5 maps to page 1

Logical address in the form of <page number, offset>

<1, 80>

Logical address: page number\*page size + offset

1\*1024 + 80= 1104

Logical address = 1104

4. (Paged Memory Management) Given the following Page-Map Table (PMT), assume the page size is 1024 and the frame size is 2048:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Page | 0 | 1 | 2 | 3 | 4 | 5 |
| Frame | 10 | 7 | 4 | 11 | 5 | 3 |

a. What is the physical address associated with the logical address of 5346?

ANS: Logical address = 5346 and page size = 1024 and frame size = 2048

Page number = address/page size

5346/1024 = 5

Offset = remainder of address over the page size

5346%1024 = 226

Logical address in the form of <page number, offset>

<5, 226>

Check PMT:

Page 5 maps to frame 3

Physical address = frame\*frame size + offset

= 3\*2048 + 226 = 6370

Physical address = 6370

b. What is the logical address associated with the physical address of 10255?

ANS: Physical address = 10255 and page size is 1024and frame size is 2048.

Frame number = address/ frame size = 10255/2048 = 5

Offset = remainder of the address over the frame size

Offset = 10255%2048 = 15

Check PMT:

Frame 5 maps to page 4

Logical address in the form of <page number, offset>

<4, 15>

Logical address: page number\*page size + offset

4\* 1024 + 15= 4831

Logical address = 4831

5. (CPU Scheduling) Given the following service times for 5 processes, draw a Gantt chart to show process scheduling using each of the following algorithms:

a. First-Come, First-Served

Total average time = (80 +180 + 240 + 280 +400)/5 = 236

80 180 240 280 400

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Service Time | 80 | 100 | 60 | 40 | 120 |
| process | P1 | P2 | P3 | P4 | P5 |

Gantt chart:

0 80 180 240 280 400

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 |

b. Shortest Job Next

Total average time = (40+ 100 + 180 +280 + 400)/5 = 200

40 100 180 280 400

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Service Time | 40 | 60 | 80 | 100 | 120 |  |
| process | P4 | P3 | P1 | P2 | P5 |

Gantt chart:

40 100 180 280 400

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| P4 | P3 | P1 | P2 | P5 |

c. Round Robin with the time slice 30

Total average time = (240 + 250 + 300 + 370 + 400)/5 = 312

400

240 250 300 370

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Service time | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 10 | 30 | 20 | 30 | 30 | 10 | 30 |
| process | P1 | P2 | P3 | P4 | P5 | P1 | P2 | P3 | P4 | P5 | P1 | P2 | P5 | P1 | P5 |

Gantt chart:

400

0 240 250 300 370

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 | P1 | P2 | P3 | P4 | P5 | P1 | P2 | P5 | P1 | P5 |

d. What is the Average Turnaround Time for each algorithm?

ANS: The Average Turnaround Time for First-Come, First-Served algorithm is 236

The Average Turnaround Time for Shortest Job Next algorithm is 200.

The Average Turnaround Time for Round Robin algorithm is 312.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Process | P1 | P2 | P3 | P4 | P5 |
| Service Time | 80 | 100 | 60 | 40 | 120 |

6. (CPU Scheduling) Given the following service times for 6 processes, draw a Gantt chart to show process scheduling using each of the following algorithms:

a. First-Come, First-Served

Total average time: (125 + 205 + 240 + 380 + 590 + 645)/6 =364 approx

0 125 205 240 380 590 645

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Service Time | P1 | P2 | P3 | P4 | P5 | P6 |
| Process | 125 | 80 | 35 | 140 | 210 | 55 |

Gantt chart

0 125 205 240 380 590 645

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 | P6 |

b. Shortest Job Next

Total average time: (35 + 90 + 170 +295 + 435 + 645)/6 = 278 approx

0 35 90 170 295 435 645

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Service Time | 35 | 55 | 80 | 125 | 140 | 210 |
| Process | P3 | P6 | P2 | P1 | P4 | P5 |

Gantt chart:

0 35 90 170 295 435 645

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| P3 | P6 | P2 | P1 | P4 | P5 |

c. Round Robin with the time slice 40

Total average time: (115+315+410+535+555+645)/6 = 429

115 315 410 535 555 645

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S T | 40 | 40 | 35 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 15 | 40 | 40 | 40 | 5 | 20 | 40 | 40 | 10 |
| P | P1 | P2 | P3 | P4 | P5 | P6 | P1 | P2 | P4 | P5 | P6 | P1 | P4 | P5 | P1 | P4 | P5 | P5 | P5 |

Gantt chart:

0 115 315 410 535 555 645

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 | P6 | P1 | P2 | P4 | P5 | P6 | P1 | P4 | P5 | P1 | P4 | P5 | P5 | P5 |

d. What is the Average Turnaround Time for each algorithm?

ANS: The Average Turnaround Time for First-Come, First-Served algorithm is 364(approx)

The Average Turnaround Time for Shortest Job Next algorithm is 278(approx)

The Average Turnaround Time for Round Robin algorithm is 429(approx)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process | P1 | P2 | P3 | P4 | P5 | P6 |
| Service Time | 125 | 80 | 35 | 140 | 210 | 55 |

7. (Disk Scheduling) Given the following list of cylinder requests in order in which they were received, assume that the disk is positioned at cylinder 25 and list the order in which these requests would be handled using each of the following Disk Scheduling algorithms:

a. First-Come, First-Served

ANS: The order will be 22, 3, 85, 14, 57, 36, 92, 46, 18, 71.

Because this algorithm handles the request in the same order as they were received.

b. Shortest-Seek-Time-First

ANS: 22, 18, 14, 3, 36, 46, 57, 71, 85, 92

It checks the difference between the adjacent values.

c. SCAN if the cylinder is moving toward the higher cylinder numbers

ANS: 36, 46, 57, 71, 85, 92,22, 18, 14, 3

Request List: 22, 3, 85, 14, 57, 36, 92, 46, 18, 71