1. Write a shell script program for different string handling functions.
2. Write a shell script program for different arithmetic functions.
3. Write a shell script program for a number to check whether it is
4. palindrome
5. prime
6. generate Fibonacci series
7. Implement the solutions for following synchronization problems using semaphore:

i. Producer Consumer

ii. Reader Writer

1. Implement the solutions for following synchronization problems using semaphore:

ii. Reader Writer

iii. Dinning Philosopher

6. Implement the solutions for following synchronization problems using semaphore:

i. Producer Consumer

iii. Dinning Philosopher

1. Write a program to implement following CPU scheduling algorithms:
2. FCFS
3. Round Robin – time slice=4ms
4. Write a program to implement following CPU scheduling algorithms:
5. SJF
6. SRTF
7. Write a program to implement following CPU scheduling algorithms:
8. Priority –preemptive
9. Priority –Nonpreemptive
10. Write a program to implement following CPU scheduling algorithms:
11. Priority –preemptive
12. FCFS
13. Write a program to implement following CPU scheduling algorithms:
14. Priority –Non preemptive
15. SJF
16. Write a program to implement following CPU scheduling algorithms:
17. Priority –Non preemptive
18. Round Robin – time slice=4ms
19. Write a program to implement following CPU scheduling algorithms:
20. SRTF
21. FCFS
22. Write a program to implement bankers algorithm

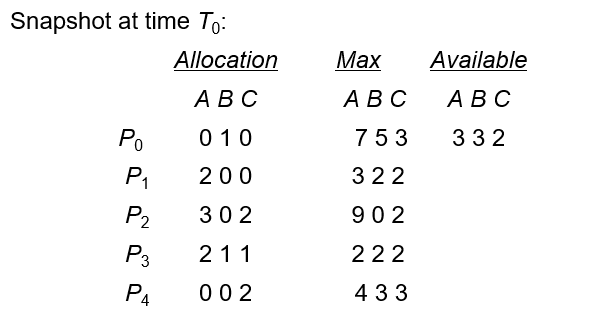
Number of processes: 5 (P0 to P4)

Resources:

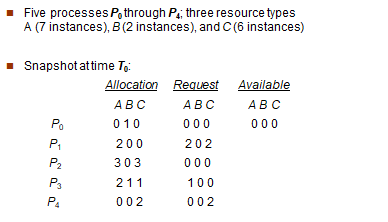
Resource A: Number of instances: 10

Resource B: Number of instances: 5

Resource C: Number of instances: 7



1. Write a program for deadlock detection algorithm for following example.



1. Following is the list of available blocks in the main memory and the process size for each process (P0 to P3).

blockSize[] = {100, 500, 200, 300, 600}

processSize[] = {212, 417, 112, 426}

Find which memory block is allocated to which process using following placement strategies:

1. First Fit
2. Worst fit c. Best fit
3. Following is the list of available blocks in the main memory and the process size for each process (P0 to P3).

blockSize[] = {100, 500, 200, 300, 600}

processSize[] = {212, 417, 112, 426}

Find which memory block is allocated to which process using following placement strategies:

1. First Fit
2. Next fit
3. Worst fit
4. Following is the list of available blocks in the main memory and the process size for each process (P0 to P3).

blockSize[] = {100, 500, 200, 300, 600}

processSize[] = {212, 417, 112, 426}

Find which memory block is allocated to which process using following placement strategies:

1. Next fit
2. Worst fit
3. Best fit
4. Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults using following page replacement algorithms:
5. First In First Out
6. Optimal
7. Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults using following page replacement algorithms:
8. Least recently used
9. Second Chance (Clock)
10. Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults using following page replacement algorithms:
11. Least recently used
12. Optimal
13. Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults using following page replacement algorithms:

1) Least recently used

2)FIFO

1. Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder using following disk scheduling algorithms:
2. FCFS
3. Shortest Seek Time First (SSTF)
4. Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder using following disk scheduling algorithms:
5. SCAN
6. C-SCAN
7. Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder using following disk scheduling algorithms:
8. SSTF
9. SCAN
10. Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder using following disk scheduling algorithms:
11. FCFS
12. SCAN
13. Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder using following disk scheduling algorithms:
14. SSTF
15. C SCAN
16. Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder using following disk scheduling algorithms:
17. FCFS
18. C SCAN
19. Implement the following algorithms:
20. FCFS- CPU scheduling algo
21. Optimal –page replacement algo –( Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults)
22. Implement the following algorithms:
23. SJF- CPU scheduling algo
24. FIFO –page replacement algo –( Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults)
25. Implement the following algorithms:
26. Round Robin- CPU scheduling algo(time slice =4ms)
27. FIFO –page replacement algo –( Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults)
28. Implement the following algorithms:
29. FCFS- Disk scheduling algo (Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder)
30. Best Fit –memory placement algo

(process (P0 to P3).

blockSize[] = {100, 500, 200, 300, 600}

processSize[] = {212, 417, 112, 426} Find which memory block is allocated to which process

1. Implement the following algorithms:
2. SSTF- Disk scheduling algo (Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder)
3. First Fit –memory placement algo

(process (P0 to P3).

blockSize[] = {100, 500, 200, 300, 600}

processSize[] = {212, 417, 112, 426} Find which memory block is allocated to which process

1. Implement the following algorithms:
2. SCAN- Disk scheduling algo (Consider the order of request for cylinder access of the disk is - (98, 183, 37, 122, 14, 124, 65, 67) and current position of Read/Write head is: 53. Find the total head movements to access the data on the cylinder)
3. Worst Fit –memory placement algo

(process (P0 to P3).

blockSize[] = {100, 500, 200, 300, 600}

processSize[] = {212, 417, 112, 426} Find which memory block is allocated to which process

1. Write a program to implement bankers algorithm

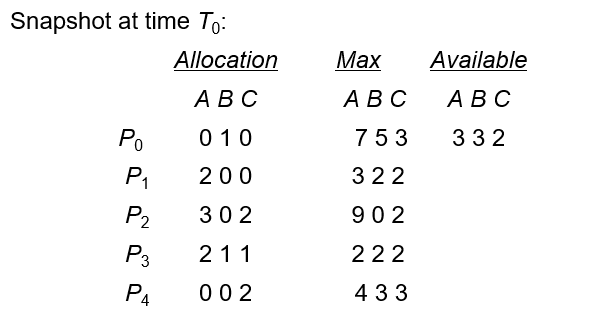
Number of processes: 5 (P0 to P4)

Resources:

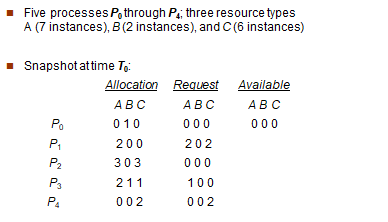
Resource A: Number of instances: 10

Resource B: Number of instances: 5

Resource C: Number of instances: 7



1. Write a program for deadlock detection algorithm for following example.



1. Write a program to implement bankers algorithm

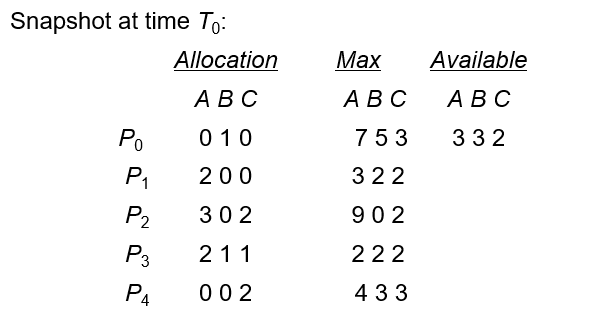
Number of processes: 5 (P0 to P4)

Resources:

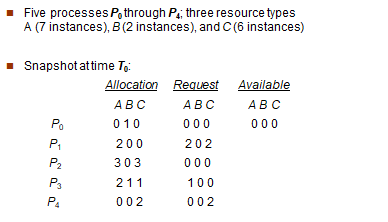
Resource A: Number of instances: 10

Resource B: Number of instances: 5

Resource C: Number of instances: 7



1. Write a program for deadlock detection algorithm for following example.



1. Following is the list of available blocks in the main memory and the process size for each process (P0 to P3).

blockSize[] = {100, 500, 200, 300, 600}

processSize[] = {212, 417, 112, 426}

Find which memory block is allocated to which process using following placement strategies:

1. Next fit
2. Worst fit
3. Best fit
4. Consider the page reference string as 7,1,0,2,0,3,0,4,2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1 with 3-page frames. Find the total number of page faults using following page replacement algorithms:
5. First In First Out
6. Optimal
7. Write a program to implement following CPU scheduling algorithms:
8. SJF
9. SRTF