Q1.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Allocation | Max | Available |
|  | A B C D | A B C D | A B C D |
| P0 | 0012 | 0012 | 1520 |
| P1 | 1000 | 1750 |  |
| P2 | 1354 | 2356 |  |
| P3 | 0632 | 0652 |  |
| P4 | 0014 | 0656 |  |

1. What is the content of the matrix Need?

|  |  |
| --- | --- |
|  | Need |
|  | A B C D |
| P0 | 0 0 0 0 |
| P1 | 0 7 5 0 |
| P2 | 1 0 0 2 |
| P3 | 0 0 2 0 |
| P4 | 0 6 4 2 |

1. Is the system in a safe state?

|  |  |  |
| --- | --- | --- |
|  |  | Available |
|  |  | A B C D |
| P0 | 1st | 1 5 3 2 |
| P2 | 2nd | 2 8 8 6 |
| P3 | 3rd | 2 14 11 8 |
| P4 | 4th | 2 2 12 12 |
| P1 | 5th | 3 14 12 12 |

Yes, it is safe.

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

The request (0, 4, 2, 0) < Need (0,7,5,0) &The request (0, 4, 2, 0) < Available(1,5,2,0)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Allocation | Max | Need | Available |
|  | A B C D | A B C D | A B C D | A B C D |
| P0 | 0012 | 0012 | 0000 | 1100 |
| P1 | 1420 | 1750 | 0330 |  |
| P2 | 1354 | 2356 | 1002 |  |
| P3 | 0632 | 0652 | 0020 |  |
| P4 | 0014 | 0656 | 0642 |  |

Now, the sequence is

|  |  |  |
| --- | --- | --- |
|  |  | Available |
|  |  | A B C D |
| P0 | 1st | 1112 |
| P2 | 2nd | 2466 |
| P3 | 3rd | 2 10 9 8 |
| P4 | 4th | 2 10 10 12 |
| P1 | 5th | 3 14 12 12 |

The state is still safe, and the request can be granted immediately.

Q3: Provide two programming examples in which multithreading provides better performance than a single-threaded solution.

1. A Web server that services each request in a separate thread.
2. An application such as matrix multiplication where each row of the matrix product is evaluated, in parallel, by a different thread.

Q4. Provide two examples where multithreading does not provide better performance than single threaded.

1. a "shell" program such as the C-shell or Korn shell. Such a program must closely monitor its own working space such as open files, environment variables, and current working directory.
2. a program that calculates an individual tax return.

Q5. Describe the differences among short-term, medium-term, and long- term scheduling.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Long term scheduler** | **Medium term scheduler** | **Short term scheduler** |
| **Type of Scheduler** | Long term scheduler is a job scheduler. | Medium term is a process of swapping schedulers. | Short term scheduler is called a CPU scheduler. |
| **Speed** | The speed of long term is lesser than the short term. | The speed of medium term is in between short and long term scheduler. | The speed of short term is fastest among the other two. |
| **Purpose** | Long term controls the degree of multiprogramming. | Medium term reduces the degree of multiprogramming. | The short term provides lesser control over the degree of multiprogramming. |
| **Minimal time-sharing system** | The long term is almost nil or minimal in the time-sharing system. | The medium term is a part of the time-sharing system. | Short term is also a minimal time -sharing system. |
| **Function** | The long term selects the processes from the pool and loads them into memory for execution. | Medium term can reintroduce the process into memory and execution can be continued. | Short term selects those processes that are ready to execute. |

Q6. Non preemptive FCFS scheduling algorithm.

* Turnaround time = Burst time + Waiting time / Turnaround time = Exit time - Arrival time

Gantt chart:

|  |  |  |
| --- | --- | --- |
| P1 | P2 | P3 |

0 8 12 13

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process | Arrival Time | Burst Time | Exit time | Turnaround time |
| P1 | 0.0 | 8 | 8 | 8 |
| P2 | 0.4 | 4 | 12 | 11.6 |
| P3 | 1.0 | 1 | 13 | 12 |
| Avg. |  |  |  | 10.53 |