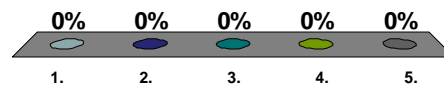


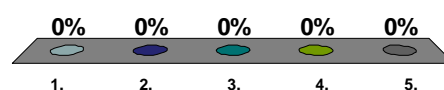
Can TFTP processes overwrite existing files
(on the client and/or server)?

1. Client only
2. Server only
3. Both Client and Server
4. Neither Client nor Server
5. Your choice



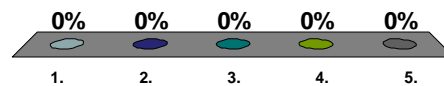
We have talked about (hard and soft) real-time systems.
What name do we give a (non-real-time) system that still
tries to do things in a timely manner?

1. Best effort
2. Interactive
3. Fake-time
4. Unreal-time
5. Active



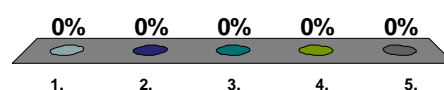
What do we call a scheduling system that permits a lower priority thread to continue using the processor until it is done with it, even if a higher priority thread comes along?

1. Preemptive
2. Priority-based
3. Anti-Preemptive
4. Deferred Preemptive
5. Non-Preemptive



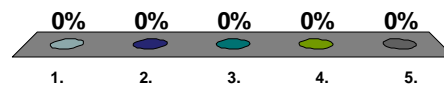
What do we call a scheduling system that permits a lower priority thread to continue using the processor for a fixed period of time, even if a higher priority thread comes along?

1. Preemptive
2. Priority-based
3. Anti-Preemptive
4. Deferred Preemptive
5. Non-Preemptive



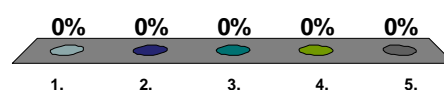
What do we call a scheduling system that does not permit a lower priority thread to continue using the processor when a higher priority thread comes along?

1. Preemptive
2. Priority-based
3. Anti-Preemptive
4. Deferred Preemptive
5. Non-Preemptive



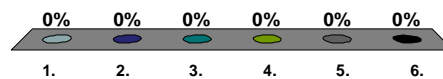
How do we know if a group of processes can be scheduled using a cyclic executive?

1. If we draw a minor/major cycle timeline and they fit
2. If they pass the Liu-Layland equation
3. If they fail the Liu-Layland equation, but we draw a timeline and they all meet their first deadline
4. All of the above
5. 2 and 3



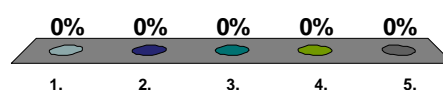
What term(s) is/are not applicable to cyclic executives?

1. Minor cycle
2. Major cycle
3. Interrupt
4. Priority
5. 3 and 4
6. 1 and 2



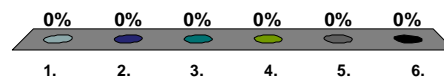
How do we know if a group of processes can be scheduled using rate monotonic priority assignment?

1. If we draw a minor/major cycle timeline and they fit
2. If they pass the Liu-Layland equation
3. If they fail the Liu-Layland equation, but we draw a timeline and they all meet their first deadline
4. All of the above
5. 2 and 3



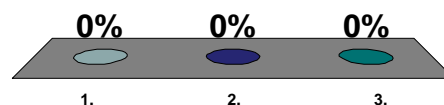
What term(s) is/are not applicable to rate monotonic scheduling?

1. Minor cycle
2. Major cycle
3. Interrupt
4. Priority
5. 3 and 4
6. 1 and 2



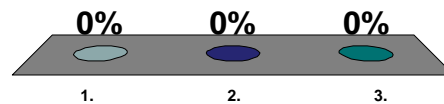
The Liu-Layland equation gives 78% as the bound for 3 processes. If the total utilization of our 3 processes is 78% then, using rate monotonic priority assignment:

1. They are schedulable
2. They are not schedulable
3. They may be schedulable



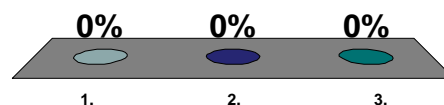
The Liu-Layland equation gives 78% as the bound for 3 processes. If the total utilization of our 3 processes is 77% then, using rate monotonic priority assignment:

1. They are schedulable
2. They are not schedulable
3. They may be schedulable



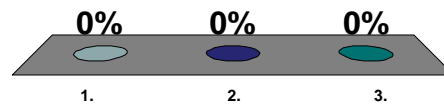
The Liu-Layland equation gives 78% as the bound for 3 processes. If the total utilization of our 3 processes is 79% then, using rate monotonic priority assignment:

1. They are schedulable
2. They are not schedulable
3. They may be schedulable



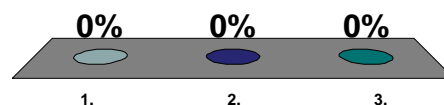
The Liu-Layland equation gives 78% as the bound for 3 processes. If the total utilization of our 3 processes is 101% then, using rate monotonic priority assignment:

1. They are schedulable
2. They are not schedulable
3. They may be schedulable



The Liu-Layland equation gives 78% as the bound for 3 processes. If the total utilization of our 3 processes is 99% then, using rate monotonic priority assignment:

1. They are schedulable
2. They are not schedulable
3. They may be schedulable



The Liu-Layland equation gives 78% as the bound for 3 processes. If the total utilization of our 3 processes is 100% then, using rate monotonic priority assignment:

1. They are schedulable
2. They are not schedulable
3. They may be schedulable

