**Quiz 2 Review**

**Waiting time** – the amount of time that the process spends waiting in the ready queue to use the processor.

**Turnaround time** – the amount of elapsed time between when a process is created and the moment that the process exits the running state by terminating.

**Response Time** – the amount of time it takes to start responding but not the time that it takes to output that response.

**Goals:**

* Maximize the throughput. (# jobs/unit of time)
* Maximize I/O utilization.
* Maximize CPU utilization.
* Minimize the response time.
* Minimize turnaround time.

**Modes of scheduling:**

**Non preemptive** – processes are allowed to remain on the CPU until they terminate, block themselves. (FCFS, Priority Scheduling, nonPreemptive SJF).

**Preemptive** – the scheduler can forcibly remove a process from the CPU.

**Questions**

1. What do you understand by a thread being in **runnable** state?

The resources are allocated to the thread. To be in ready or run state

1. What are the advantages of implementing threads versus regular processes?

Multithreading are faster. Threads use minimal resources.

1. In what state a thread goes if it executes **yield()?**

The thread goes to runnable state.

1. What **scheduling algorithm** Java threads are using?

Java threads use preemptive priority scheduling system.

1. What is the **default(norm)** priority of a Java thread?

Default priority is 5 in java thread

1. The **sleep(t)** method needs to be inside of a **try{}…catch{}** block. Explain why.

If the thread that is interrupted is blocked, the method that blocked the thread throws an InterruptException object.

1. In what units **t** (of sleep(t)) is expressed?

Milliseconds

1. What are the two ways of providing the **run()** method?
2. Subclassing the thread class and overriding the run() method
3. Implementing the Runnable interface.
4. What method will invoke the **run()** method?

Start()

1. What is the outcome of executing **t2.join()**?

It will wait for thread t2 to die.

1. Mention **two methods** of the object class used for **strong synchronization** of threads.
2. Wait()
3. Notify()
4. NotifyAll()