(1) How can “race condition” happen? Show “how” using an example.

A race condition can happen when more than one process updates the same data (in shared memory). As shown in the following example in assembly program, unpredictable context-switching can happen when a process (P₁) loads the contents of the shared memory, P₁ can be preempted to another process (P₂), which starts executing the three instructions in the example.

As a result, there are three possible results for “A”, 5, 8 and -3. That is how a race condition can happen.

(2) What is “critical section”?

Critical section is a portion of binary executable programs that cause race condition.
(3) What is “mutual exclusion”? 

Mutual exclusion means “only one process at a time (when a process is present, no other process can be there)”. It is a solution for race condition.

(4) What does “atomic” in “atomic operations” mean?

The term, “atomic”, means “once an operation (a task) starts, it will not be interrupted (it goes to its end without interruption).”

(5) What “signal” system call to a semaphore exactly performs?

Signal

- If no one waiting on S, do S = S + 1
- If some one waiting on S, let the first proceed to CS and leave S = 0