1. Explain why the monitor <ARMON> prompt appears between the prints for 5 and 4.

Since I use if() loop to check the counter value and decrement it, an ARMON appears between 5 & 4. If I have used while() loop instead of that, it will not display ARMON in between.

1. We are using shared memory for communication here, why do we not need a semaphore, or other mutex operation to protect accesses to the shared variable?

Shared Memory passes data between the programs. Shared memory is a space where two processes sharing a common space in memory in which they read and write from for communication between these two spaces. Shared memory is the fastest IPC mechanism. If we know where the shared area of memory, it is like other part of the address space of processor.

Mutexes are other IPC synchronization mechanisms. Mutex is mutual exclusion object. It is a program object which allows multiple program threads to share the same resource, like file access. But it cannot be used to access different files simultaneously using this mechanism.

Semaphores is another IPC mechanism. It is used in such a way that it will allow different processes synchronize access to shared resources. It is not used to exchange large amount of data. They are used for synchronization among processes.