

Worksheet 3.3: Processes, Multiple-Choice Questions

In the questions below, circle the right answer. There is only one correct answer.

1. Which of the following is **not** true about a **Zombie** process?
- a. It has an entry in the process table.
 - b. Its parent has terminated without calling wait().
 - c. Its parent has not called wait(), but the parent has not terminated yet.
 - d. Its resources have been deallocated by the operating system.

Explanation: A zombie process is a process that has completed execution but still has an entry in the process table because its parent has not yet called wait() to read its exit status. If the parent terminates without calling wait(), the child process typically gets adopted by the init process, which will then call wait() to remove the zombie process.

2. What is the difference between shared memory and message passing?
- a. The format of the shared memory object is determined by user processes while the format of a message is determined by the OS.
 - b. A shared memory block is physically located in the user address space while mail boxes are in the kernel address space.
 - c. Shared memory communication involves more kernel intervention than message passing.
 - d. Shared memory is faster than message passing on all systems.
 - e. Both a and b are correct.
 - f. Both b and c are correct.
 - g. a, b and c are correct.

Explanation: In shared memory, the user processes determine the format of the shared memory objects, and shared memory blocks are typically located in the user address space. Mailboxes used in message passing are generally located in the kernel address space. Shared memory usually involves less kernel intervention compared to message passing.

3. Which of the following is **not** true about Remote Procedure Calls (RPCs)?
- a. The matchmaker takes an RPC name and returns the corresponding port number.
 - b. The user program communicates directly with the matchmaker without kernel intervention.
 - c. Parameter marshalling involves packaging function parameters in a form that can be transmitted over a network.
 - d. Parameter marshalling is done by the kernel.

Explanation: The matchmaker typically operates with kernel intervention to ensure security and proper communication. RPC parameter marshalling packages function parameters for network transmission, but this task is usually handled by the user-level libraries, not directly by the kernel.

4. What's the difference between blocking receive and non-blocking receive?
- a. With blocking receive, a process is put in the waiting state if no message is available.
 - b. With non-blocking receive, a process is put in the waiting state if no message is available.
 - c. Blocking receive must be used if the send is blocking.
 - d. A non-blocking receive may not return an actual message, while a blocking receive always returns a message (unless there is an error).
 - e. Both b and d are correct.
 - f. Both a and d are correct.
 - g. a, c and d are correct.

Explanation: In a blocking receive, the process waits (is put in the waiting state) if no message is available, ensuring it always gets a message unless there's an error. In a non-blocking receive, the process continues execution even if no message is available, and it may not return an actual message.