Quiz on Threads in Distributed Systems

Please select the correct option for each question.

Questions

1. What is a **pop-up thread** in the context of distributed systems?
   - ☐ A) A thread that runs in the background
   - ☐ B) A thread created only when a request arrives
   - ☐ C) A thread that is pre-allocated in memory
   - ☐ D) A thread that handles only I/O operations
2. What is a primary advantage of using pop-up threads?
   - ☐ A) They can run indefinitely
   - ☐ B) They have a longer execution time
   - ☐ C) The latency between request arrival and processing start is very short
   - ☐ D) They require less memory than regular threads
3. In which space is it generally faster to create a pop-up thread?
   - ☐ A) User space
   - ☐ B) Kernel space
   - ☐ C) Virtual space
   - ☐ D) Shared memory space
4. What is a potential downside of creating threads in the kernel space?
   - ☐ A) It consumes more CPU resources
   - ☐ B) A bug in a kernel thread can lead to severe system problems
   - ☐ C) It is more difficult to manage user data
   - ☐ D) It requires additional configuration
5. Why might using pop-up threads be beneficial in server applications?
   - ☐ A) They increase the server's uptime
   - ☐ B) They can handle requests more efficiently
   - ☐ C) They reduce the need for inter-thread communication
   - ☐ D) They eliminate the need for locking mechanisms
6. When a pop-up thread is created, what does it lack?
   - ☐ A) The ability to communicate with other threads
   - ☐ B) Any history (registers, stack, etc.) to restore
   - ☐ C) Access to I/O devices
   - ☐ D) Memory allocation
7. What planning consideration is necessary when implementing pop-up threads?
   - ☐ A) The maximum number of threads allowed
   - ☐ B) The scheduling policy of the operating system
   - ☐ C) The location of thread creation (user/kernel space)
   - ☐ D) The duration of thread execution
8. In distributed systems, what often causes threads to remain blocked?
   - ☐ A) Waiting for a request to be processed
   - ☐ B) Memory allocation issues

☐ C) Lack of CPU resources
☐ D) Networking delays

Answers

1. B) A thread created only when a request arrives
2. C) The latency between request arrival and processing start is very short
3. B) Kernel space
4. B) A bug in a kernel thread can lead to severe system problems
5. B) They can handle requests more efficiently
6. B) Any history (registers, stack, etc.) to restore
7. C) The location of thread creation (user/kernel space)
8. A) Waiting for a request to be processed