Multithreading in C: Understanding and Managing Threads Quiz

1. What is the main benefit of using threads over processes?

- ☐ A) Threads provide more isolation between tasks
- ☒ B) Threads share memory space, making them more efficient
- ☐ C) Threads are easier to debug than processes
- ☐ D) Threads do not require synchronization mechanisms

2. Which library must be included to use POSIX threads in C?

- ☐ A) `stdio.h`
- ☒ B) `pthread.h`
- ☐ C) `stdlib.h`
- ☐ D) `unistd.h`

3. What is the correct syntax to create a thread in C using pthreads?

- ☐ A) `pthread_create(&tid, thread_function, NULL);`
- ☒ B) `pthread_create(&tid, NULL, thread_function, argument);`
- ☐ C) `pthread_create(tid, NULL, argument, thread_function);`
- ☐ D) `pthread_create(tid, thread_function, &argument);`

4. Which function returns the thread ID of the calling thread?

- ☐ A) `pthread_id()`
- ☐ B) `pthread_join()`
- ☒ C) `pthread_self()`
- ☐ D) `pthread_get_id()`

5. What happens when a thread calls `pthread_exit()`?

- ☐ A) The thread continues execution
- ☒ B) The thread is terminated
- ☐ C) The thread is suspended until resumed
- ☐ D) The thread ID is returned to the main thread

6. Which function makes the calling thread wait for another thread to terminate?

- ☐ A) `pthread_wait()`
- ☐ B) `pthread_cancel()`
- ☐ C) `pthread_exit()`
- ☒ D) `pthread_join()`

7. What is the purpose of `pthread_detach()`?

☐ A) To terminate a thread immediately
☒ B) To mark a thread as detached, releasing resources upon termination
☐ C) To cancel the execution of a thread
☐ D) To wait for a thread to complete

8. In a multithreading context, what is a race condition?

☐ A) Two threads executing the same code block at the same time
☐ B) A thread terminating before its task completes
☒ C) Multiple threads accessing shared data without proper synchronization
☐ D) A thread that outperforms another thread in execution time

9. What does `pthread_cancel()` do?

☐ A) It waits for a thread to terminate
☒ B) It sends a cancellation request to a thread
☐ C) It terminates a thread immediately
☐ D) It returns the result of a thread's execution

10. Which of the following correctly describes a thread?

☐ A) An independent process with its own memory space
☒ B) A lightweight process sharing memory space with other threads
☐ C) A static task that runs sequentially
☐ D) A dynamically allocated data structure

11. What mechanism ensures that shared resources are accessed safely by multiple threads?

☐ A) Thread creation
☒ B) Synchronization
☐ C) Memory allocation
☐ D) Thread detachment

12. What does the `pthread_join()` function return when a thread finishes normally?

☐ A) `NULL`
☐ B) The thread ID
☒ C) The thread's status
☐ D) The thread's memory address

13. What is the main reason for using threads in C?

☐ A) Improved readability
☐ B) Increased complexity in managing tasks

☒ C) Improved resource utilization and efficiency
☐ D) Easier debugging of code

14. Which function allows threads to execute concurrently in a C program?

☒ A) `pthread_create()`
☐ B) `pthread_run()`
☐ C) `pthread_execute()`
☐ D) `pthread_compete()`

15. What is thread safety?

☐ A) Ensuring that threads do not terminate unexpectedly
☐ B) Allowing threads to run independently without synchronization
☒ C) Protecting shared resources from being accessed concurrently in an unsafe manner
☐ D) Guaranteeing that all threads run in the correct order

Answers:

1. B
2. B
3. B
4. C
5. B
6. D
7. B
8. C
9. B
10. B
11. B
12. C
13. C
14. A
15. C