

Q1. What resources are used when a thread is created? How do they differ from those used when a process is created?

A: When a thread is created, there are some resources being used: a register set location for storage for context switching; a local stack to record the procedure call arguments, return values, return addresses; thread-local storage. Since thread is smaller than process, these resources are fewer than process creation. Creating a process requires a new address space, a process control block (PCB), which is a very large data structure. Thus, creating a thread is less expensive than creating a process.

Q2. Is it possible to have concurrency but not parallelism? Explain.

A: Yes, it is.

Concurrency: two different tasks or threads start working together at the same time period, however, it does not mean they run at same instant.

Parallelism: two or more different tasks start their execution at the same time. It means that the two tasks or threads start working simultaneously, and is carried by scheduling task.

Concurrency often be seen on multi-task switch on a single-processor while parallelism can be seen on multi-cored processor.

Concurrency can occur if the process switching is fast without the need of parallelism.