-"Spawn" and "sync" are two key terms used to describe parallel computing in the CLRS textbook.

"Spawn" indicates the launch of a new thread or process to perform a task. In parallel algorithms, if a task can be divided into multiple subtasks, these subtasks can be assigned to different threads or processes to execute. "Spawn" refers to the operation of starting these threads or processes.

* "Sync" represents a synchronization operation, which means waiting for multiple threads or processes to complete before proceeding with the next operation. In parallel algorithms, if an algorithm needs to wait for multiple threads or processes to complete before proceeding to the next step, "sync" operation is required.

-Using Multithreaded merge sort has the following advantages over regular merge sort:

1. Faster sorting speed: Multithreaded merge sort can take advantage of multi-core CPUs by dividing the array into multiple sub-arrays and sorting them simultaneously in different threads. When merging the sorted sub-arrays, only the already sorted sub-arrays need to be merged, which can reduce the sorting time, especially when sorting large arrays.
2. Higher scalability: Since Multithreaded merge sort can be executed in parallel, it can easily be extended to handle larger datasets, thus achieving better scalability.
3. Higher concurrency: Multithreaded merge sort can perform multiple sorting operations simultaneously, thus achieving higher concurrency and better response time.

In terms of time complexity, both Multithreaded merge sort and regular merge sort have a time complexity of O(nlogn). However, Multithreaded merge sort can be faster than regular merge sort when processing large datasets because it can utilize multiple CPU cores simultaneously.

Furthermore, Multithreaded merge sort can avoid the cache invalidation problem caused by moving large amounts of data in memory, further improving performance.