40826416 資工二乙 陳筱婷

1. Concurrency: Support more than one task making process

Parallelism: Imply a system can perform more than one task simultaneously.

2. ① Identifying task ② Balance ③ Data splitting ④ Data dependency ⑤ Testing & Debugging

3. Formula:

$$\text{Speedup} \leq \frac{1}{S + (\frac{1-S}{N})}$$

Indentify performance gains from adding additional cores to an application that has both serial and parallel component.

S: partion of the application that must be performed serially on a system.

N: processing code.

4. ① Asychronous threading

a. Parent resumes its execution after generating child thread.

b. Run parent and child concurrently and independently.

c. Little data sharing

② Sychronous threading

a. Parent waits for all of its children threads to terminate.

b. Children threads run concurrently

c. Significant data sharing

5. Multilevel queue scheduling

→ Ready queue is sperated into two parts – foreground and background.

→ Each queue has its own scheduling algorithm.

a. Foreground – RR

b. Background – FCFS

5. Difference

a. In MLFQ, process can move between the various queues, and MLQ can't do that

b. In MLFQ, the queues are classified as higher priority queue and lower queue, and this can avoid starvation. In MLQ, the priority can't change, so may cause starvation

6. Convoy effect - short process behind long process

7. ① Knowing the length of the next CPU request

② Estimate the length - should be similar to the previous one

8. ① Problem ≡ Starvation (indefinite blocking) - Low priority processes may never execute.

② Solution ≡ Aging - as time progressess increase the priority of the process.