**Threads**

When threads start and pause, in the same blocks as other threads, this is called interleaving.

Interleaving can be done when instructions from multiple threads are mixed together during execution in unpredictable order.

It happens because a CPU can **switch rapidly** between threads, and unless operations are atomic, they can **overlap** in weird ways.

In Programming, an atomic action is one, that effectively happens all at once.  
atomic action is operation that is Indivisible - it either happens completely or doesn’t happen at all, and no other thread can see it partially done or interfere with it.

* Indivisible – can not be broken into smaller parts or interrupted
* Uninterrupted – No other thread can interleave during the action
* Consistent – the system sees only “before” or “after” state, never “in-between”