3:

* Concurrency – two or more tasks can start, run and complete in overlapping time periods, (multitasking on a single-core machine). It doesn’t mean they’ll ever both be running at the same instant, which would be parallelism, (a multicore processor).
* To increase speed without increasing the clock speed.
* Problems where you need several tasks running simultaneously, e.g. elevators
* Harder
* Process – one or more threads of execution, these threads are executed concurrently

Thread – smallest sequence of instructions which can be managed independently.

Green thread – Denne var litt verre. Threads on the user level. They can be used to simulate multi-threading on platforms that don’t provide that capability. (Vennligst forklar litt nærmere studass)

Coroutines –

* Which one of these do pthread\_create() (C/POSIX), threading.Thread() (Python), go (Go) create?

pthread\_create() – creates a thread

threading.Thread() – creates a thread

go – co-routine

* GIL – prevents multiple threads running at the same time