Moore's Law

Moore's Law says that transistor density doubles (roughly) every two years. The reason is that transistors become smaller, and therefore more transistors fit into the chip. The access to memory, however, does not become faster at the same rate, which is known as the Von Neumann bottleneck. Thus, in order to have program speed scale as processors become faster, processor designers started to put very fast memory chips (cache memory) inside the processor chip.

Recently, Moore's Law stopped holding true, for several reasons.
The first is that, as transistors become smaller, **transistor density increases** in the processor (there are more transistors in the same area). **More transistors consume more power and also produce more heat**, which in turn poses problems for efficient cooling.

In theory, as the size of transistors decreases, their voltage should also decrease (Dennard Scaling), i.e. *voltage scaling reduces (dynamic) power consumption*. However, voltage cannot go too low, it must stay above a certain threshold. Besides, Dennard Scaling does not consider leakage power in the processor. **In other words, voltage scaling cannot prevent leakage power loss**, and **voltage scaling is limited due to noise or threshold voltage**. Thus, the power consumption is still a problem for modern processors.

As a consequence of Moore's Law not holding anymore, processor designers have to find other ways to increase processor power. One thing they do overcome these limitations is to create more cores inside the processor, even if the frequency of each core is not higher than its predecessor. But these multi-core systems require different programming models, namely parallel and concurrent programming.