

# Why Parallel Programming?

We now enter the world of parallel programming, a modern technique related to multitasking. Let's examine why it's so essential.

Simply put, because it may speed up your code. Unlike 10 years ago, today, your computer (and probably even your smartphone) has one or more CPUs that have **multiple processing cores** ([Multi-core processor](#)). This helps with desktop computing tasks like multitasking (running multiple programs, plus the operating system, simultaneously). For scientific computing, this means you have the ability in principle of splitting up your computations into groups and running each group on its own processor.

Most operating systems have a utility that allows you to visualize processor usage in real-time. Mac OSX has “Activity Monitor”, Gnome/GNU Linux has “gnome-system-monitor” and Windows has ... well actually I have no idea, you're on your own with that one. Fire it up, and run a computationally intensive program you have written, and what you will probably see is that you have a lot of computational power that is sleeping. Parallel programming allows you in principle to take advantage of all that dormant power.

It is obvious that parallel programming is a very powerful tool, especially with the degree of freedom it provides. However, there are certain types of parallel programming which fit different situations. We'll look at these types next.