



Computer Science 335
Parallel Processing and HPC
Siena University
Fall 2026

Why Parallel Processing?

The basic idea of parallel computing is simple enough – when one computer processor isn't powerful enough to solve the problem you want to solve in the timeframe in which you want that solution, use more than one!

Let's start by thinking about a very straightforward problem:

Example Problem:

Adding up a collection of numbers (*e.g.*, sum of an array)

Example: Taking a Census of Loudonville

One person doing this would visit each house, count the people, and ask whatever questions are supposed to be asked. This person would keep running counts. At the end, this person has gathered everything.

If there are two people, they can work concurrently. Each visits some houses, and they need to “report in” along the way or at the end to combine their information. But how to split up the work?

Example: Grading a Stack of Exams

Suppose each has several questions. Again, assume two graders to start.

Key Terms So Far

master-worker/bag of tasks

domain decomposition

pipeline

What might work well for adding two $1,000,000 \times 1,000,000$ matrices?

In each of these examples, we have taken what we might normally think of as a _____ process, and taken advantage of the availability of _____ to make use of multiple workers (in computing, the work will be done by cores or processing units).

Parallelism adds complexity (as we will see in great detail), so why bother?

- *speedup*
- *scale-up*
- *convenience*
- hardware trends