Top-Down vs Bottom-Up

In this lesson, we will look at the pros and cons of top-down dynamic programming and bottom-down dynamic programming.

We'll cover the following

Comparison

Both top-down dynamic programming and bottom-up dynamic programming have their own pros and cons. Knowing these pros and cons can help you choose the algorithm's design approach based on your problem's requirements.

Operation	Top-Down	Bottom-Up
Ease of problem formulation	✅ It is easier to formulate a problem in top-down dynamic programming because of its recursive nature	Thinking about a problem in a bottom-up manner can be slightly less intuitive
Stack management	Stack memory can quickly explode in top-down algorithms, thus making them less practical for larger inputs	✅ In bottom-up algorithms, stack memory is never an issue because there are no recursive calls
Cost of recursion	Recursive calls can entail a lot of computation cost	✅ There is no excessive computation cost due to recursive calls

Code readability	✅ Top-down algorithms are typically much easier to read and comprehend	Bottom-up algorithms can sometimes be difficult to grasp
Subproblems solved	✅ In top-down algorithms, we only evaluate a result on ad-hoc basis, i.e., when it is needed, meaning results that are not needed are never evaluated	Since we evaluate results in order starting from smallest to the largest, we may end up evaluating results that are not needed

By no means does this table mean that either of the approaches is better than the other. In fact, your choice of approach largely depends on your use case. If efficiency is more important than code readability, you may opt for a bottom-up approach or vice versa.

In the next lesson, we will quiz your knowledge of bottom-up dynamic programming.