educative

LEARN A CODING INTERVIEW PATTERN: DYNAMIC PROGRAMMING

educative

HOW DOES DYNAMIC PROGRAMMING WORK?

Many problems are solved using a divide-and-conquer approach recursively.

In these problems, we see an optimal substructure, i.e., the solution to a smaller problem helps us solve the bigger one.

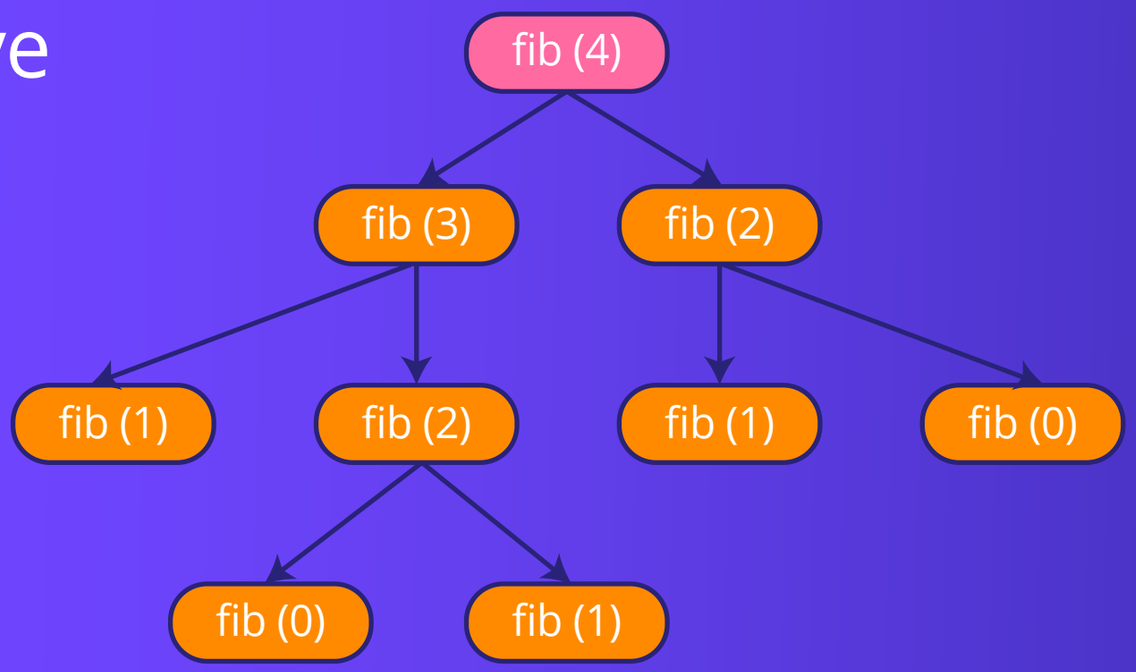
In some of the problems that can be solved with the above approach, there are many overlapping sub-problems. That is, we find ourselves solving the same sub-problem over and over.

Let’s look at an example on the next slide.

educative

HOW DOES DYNAMIC PROGRAMMING WORK?

An example is the recursive computation of the nth Fibonacci number. Here's the recursion tree for the solution to this problem with n = 4.



We can see above that fib(2) is evaluated twice, fib(1) is evaluated thrice, and fib(0) is evaluated twice. These are repeated overlapping subproblems.

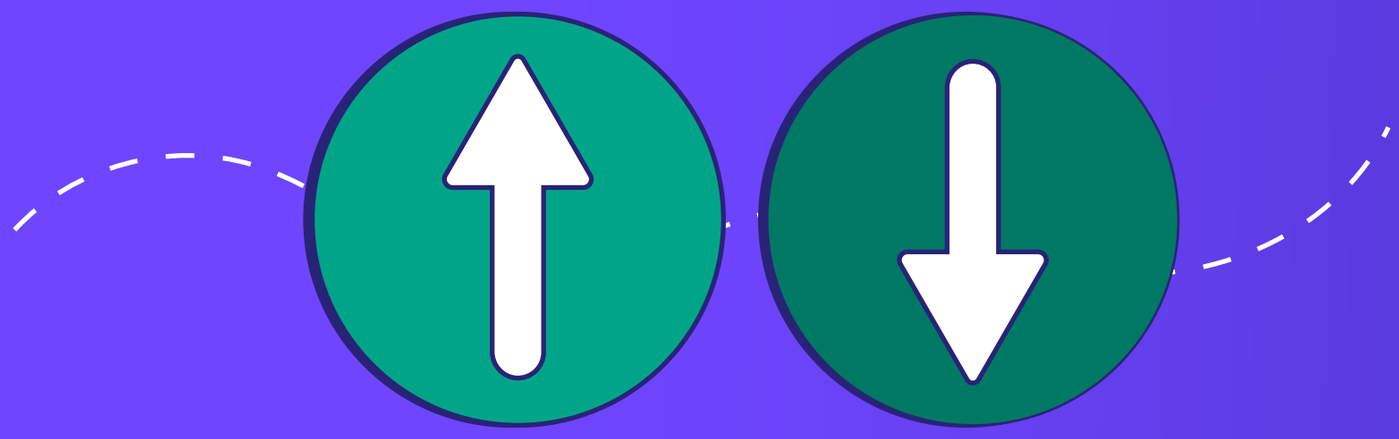
The dynamic programming pattern is used to store this result and utilize it whenever the subproblem is repeated.

educative

THE TWO MAIN APPROACHES OFDYNAMIC PROGRAMMING:

- **Top-down approach:** A recursive approach that stores the results of redundant function calls to avoid repeating calculations for the same subproblems.

- **Bottom-up approach:** An iterative strategy that systematically fills a table with subproblem results to solve larger problems efficiently.



educative

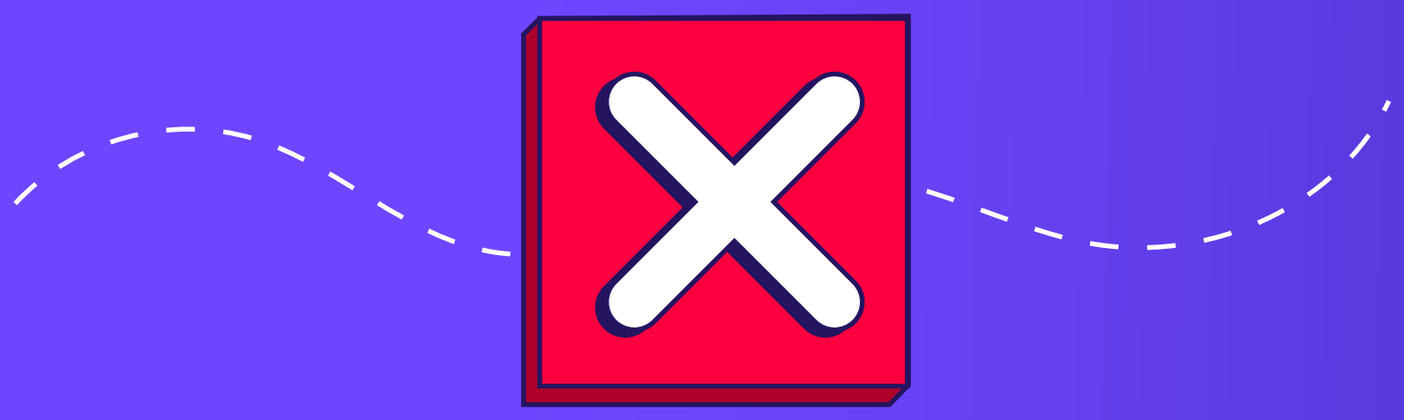
USE DYNAMIC PROGRAMMING:

- If your coding problem has overlapping subproblems.
- If your coding problem has optimal substructure.

educative

DON'T USE DYNAMIC PROGRAMMING:

- If your problem has non-overlapping subproblems.
- If your problem violates the optimal substructure property.



educative

REAL-WORLD APPLICATIONS OF DYNAMIC PROGRAMMING:

- **Optimal route planning in GPS navigation systems:** Iteratively explores all possible paths to determine the optimal route that minimizes travel time.

- **Robotics motion planning:** Explores different paths and evaluates the associated costs, determining the optimal trajectory.

educative

KEEP GOING!

Test out your new knowledge of Dynamic Programming by solving the popular 0/1 Knapsack problem!

Happy learning!