

# Chapter 5

## Dynamic Programming With Python


**Gurram Sunitha**

*School of Computing, Mohan Babu University, Tirupati, India*

**Arman Abouali**

*Technical University Clausthal, Germany*

**Mohammad Gouse Galety**

 <https://orcid.org/0000-0003-1666-2001>

*Samarkand International University of Technology, Uzbekistan*

**A. V. Sriharsha**

*Mohan Babu University, India*

### ABSTRACT

*Algorithms are at the heart of computer programming. They form the basis of all software applications and help to solve complex computational problems. Various problem-solving strategies involve divide and conquer, greedy, recursion, dynamic programming, backtracking, etc. This can be used to solve optimization problems that have overlapping subproblems. It aims to find an optimal solution by breaking down a problem into sub-problems in order to manage the complexity of the problem-solving and remembering their solutions in order to avoid repeating computation time in future steps. Mathematical optimization is a crucial component of dynamic programming, and it allows us to efficiently solve a wide range of problems that involve making decisions over time. This chapter discusses dynamic programming's relevance, mathematical optimization, ability to solve a wide range of issues, important qualities, and top-down and bottom-up problem-solving methodologies. Dynamic programming solves some typical computational problems effectively, and Python code is supplied for reference.*

DOI: 10.4018/978-1-6684-7100-5.ch005

## **INTRODUCTION**

Algorithms are at the heart of computer programming. They form the basis of all software applications and help to solve complex computational problems. Various problem-solving strategies involve divide & conquer, greedy, recursion, dynamic programming, backtracking, etc. In optimization problems, greedy algorithms shall be used to find the best outcome given a set of conditions. The backtracking algorithm is used for problems that require exploring multiple solutions before finding the best one. The divide & conquer technique breaks a problem into smaller parts so that each part can be solved separately before being combined for a final solution. Finally, dynamic programming aims to find an optimal solution by breaking down a problem into sub-problems to manage the complexity of the problem-solving and remembering their answers to avoid repeating computation time in future steps. This chapter explores the fundamentals of dynamic programming and presents a detailed discussion of various computational problems solved using dynamic programming. Also, Python code is provided for reference.

## **DYNAMIC PROGRAMMING**

Dynamic Programming is an essential algorithmic technique used to solve complex problems efficiently. It is widely used in various fields, such as computer science, operations research, and economics. Dynamic programming can solve optimization problems with overlapping subproblems by breaking them down into smaller subproblems and storing their solutions in a table for later use. This technique helps reduce the time complexity of these problems and makes them easier to solve (Kool et al., 2022). Dynamic programming is beneficial when dealing with issues that depend on previously calculated solutions or where multiple decisions must be made simultaneously (Şenaras, et al., 2021).

“Dynamic Programming” makes sense as dynamic means ‘changing’ and programming means ‘the set of instructions to be executed.’ This was introduced by “Richard Bellman” in the 1950s, which is a mathematical optimization and a computer programming technique. This has been used in various disciplines, from economics to aeronautical engineering. Let us imagine what happens if a non-optimized code consumes much memory and computing time. Consequently, the processor may need help allocating resources for other services. Most of these problems can be controlled by using dynamic programming (Bertsekas, 2022).

Expressed, this is a technique for resolving complex problems by breaking them down into smaller, more manageable sub-problems and archiving the solutions for use when the smaller sub-problems come up later. The subproblems are computed

19 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: [www.igi-global.com/chapter/dynamic-programming-with-python/326080](http://www.igi-global.com/chapter/dynamic-programming-with-python/326080)

## Related Content

---

### File Processing

(2023). *Principles, Policies, and Applications of Kotlin Programming* (pp. 241-304).

[www.irma-international.org/chapter/file-processing/323939](http://www.irma-international.org/chapter/file-processing/323939)

### Array Processing

(2023). *Principles, Policies, and Applications of Kotlin Programming* (pp. 147-158).

[www.irma-international.org/chapter/array-processing/323936](http://www.irma-international.org/chapter/array-processing/323936)

### Big Data Analytics for Geospatial Application Using Python

Assefa Senbato Genale (2024). *Ethics, Machine Learning, and Python in Geospatial Analysis* (pp. 254-278).

[www.irma-international.org/chapter/big-data-analytics-for-geospatial-application-using-python/345912](http://www.irma-international.org/chapter/big-data-analytics-for-geospatial-application-using-python/345912)

### Empowering Scientific Computing and Data Manipulation With Numerical Python (NumPy)

Tesfaye Fufa Gedefa, Galety Mohammed Gouseand Garamu Tilahun Iticha (2023). *Advanced Applications of Python Data Structures and Algorithms* (pp. 147-161).

[www.irma-international.org/chapter/empowering-scientific-computing-and-data-manipulation-with-numerical-python-numpy/326082](http://www.irma-international.org/chapter/empowering-scientific-computing-and-data-manipulation-with-numerical-python-numpy/326082)

### Technology in Education for People with Disabilities: Coding for Inclusion and Accessible Learning Design

Abhinav Varshney, Anushka Samaiya, Shrinu Varshney, Kunj Bihari Meenaand Vipin Tyagi (2026). *Effective Coding Skill Development in Education* (pp. 253-290).

[www.irma-international.org/chapter/technology-in-education-for-people-with-disabilities/398441](http://www.irma-international.org/chapter/technology-in-education-for-people-with-disabilities/398441)