

**THE BUBBLE SORT METHOD FOR ARRAY SORTING IN PYTHON
PROGRAMMING**

Fathiddinov Saidma'ruf Lazizxo'ja o'g'li
*Department of General Technical
Sciences, Asia International University*

Abstract: This article presents a comprehensive overview of the bubble sort method for array sorting in Python programming, its algorithmic structure, functionality, and practical applications. Sorting algorithms play a crucial role in data processing, optimization, and software development. Bubble sort is one of the simplest and most intuitive sorting methods, widely used for educational purposes and small-scale applications. The study highlights the importance of understanding basic sorting algorithms as a foundation for mastering more complex techniques. The results demonstrate that although bubble sort is not the most efficient algorithm for large datasets, it remains valuable for learning algorithmic thinking and basic data manipulation in Python.

Keywords: Python, bubble sort, sorting algorithms, arrays, lists, data structures, programming.

Introduction

In modern programming, data organization and processing are essential tasks. One of the most common operations performed on data collections is sorting. Sorted data improves the efficiency of searching, filtering, and analyzing information in software systems. Therefore, sorting algorithms are a fundamental topic in computer science and programming education.

Bubble sort (pufakcha metodi) is one of the simplest sorting algorithms. Its working principle is based on repeatedly comparing adjacent elements of an array and swapping them if they are in the wrong order. Although this method is not efficient for large datasets, it is easy to understand and implement, which makes it popular in introductory programming courses.

The main objective of this article is to analyze the bubble sort method in Python, explain its algorithmic principles, and evaluate its practical significance in programming education and small-scale applications.

Research Methods

This study is based on theoretical and practical analysis of sorting algorithms in Python, with a focus on the bubble sort method. At the first stage, computer science textbooks, algorithm theory resources, and Python documentation were reviewed to identify the fundamental characteristics of bubble sort.

At the second stage, a comparative approach was applied to analyze bubble sort in relation to other sorting methods such as selection sort and built-in Python sorting functions. Simple experimental examples were considered to evaluate the performance of bubble sort on small and medium-sized datasets.

Logical analysis, induction, and deduction methods were used to generalize the advantages and limitations of the bubble sort algorithm in Python programming.

Results

The results show that bubble sort is an intuitive and easy-to-implement algorithm for sorting arrays (lists) in Python. The algorithm works by repeatedly traversing the list and swapping adjacent elements if they are in the incorrect order. This process continues until the list becomes fully sorted.

Bubble sort is particularly useful for educational purposes, as it helps beginners understand fundamental concepts such as loops, conditional statements, comparisons, and data manipulation. For small datasets, bubble sort performs adequately and produces correct results.

However, the study confirms that bubble sort has low performance for large datasets due to its quadratic time complexity. As the size of the array increases, the number of comparisons and swaps grows significantly, which leads to slower execution compared to more advanced algorithms.

Discussion

The findings confirm that the bubble sort method plays an important role in teaching basic algorithmic concepts in Python programming. Its simple logic makes it suitable for illustrating how sorting works at a fundamental level. However, in real-world applications involving large volumes of data, bubble sort is rarely used due to its inefficiency.

Compared to Python's built-in sorting mechanisms and advanced algorithms, bubble sort requires more computational time and resources. Nevertheless, it remains a valuable tool for learning purposes and for solving simple problems where performance is not critical.

Another important aspect is algorithmic thinking. By studying bubble sort, programmers develop a deeper understanding of how data structures can be manipulated step by step, which serves as a foundation for learning more efficient sorting techniques.

Conclusion

In conclusion, the bubble sort (pufakcha) method is a simple yet important sorting algorithm in Python programming. Although it is not suitable for performance-critical and large-scale applications, it plays a significant role in programming education and in developing basic algorithmic skills. Understanding bubble sort helps programmers build a strong foundation for mastering more complex and efficient sorting algorithms in the future.

References

1. Millman, K. J. & Aivazis, M. Python for scientists and engineers. *Comput. Sci. Eng.* 13, 9–12 (2011).
2. Oliphant, T. E. Python for scientific computing. *Comput. Sci. Eng.* 9, 10–20 (2007).
3. Dubois, P. F., Hinsen, K. & Hugunin, J. Numerical Python. *Comput. Phys.* 10, 262–267 (1996).
4. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830 (2011).
5. Ascher, D., Dubois, P. F., Hinsen, K., Hugunin, J. & Oliphant, T. E. An Open Source Project: Numerical Python (Lawrence Livermore National Laboratory, 2001).
6. Oliphant, T. E. Guide to NumPy 1st edn (Trelgol Publishing, 2006).
7. Dubois, P. F. Python: batteries included. *Comput. Sci. Eng.* 9, 7–9 (2007).