

Federal University of Ouro Preto
PCC104 - Project and Analysis of Algorithms
Brute Force and Exhaustive Search - Part 2

Prof. Rodrigo Silva

October 22, 2024

1 Recommended Reading

- Chapter 3 - *Introduction to the Design and Analysis of Algorithms (3rd Edition)* - Anany Levitin
- Book - *Problem Solving with Algorithms and Data Structures using C++* (available at: <https://runestone.academy/runestone/books/published/cppds/index.html#>)
- Arrays <https://www.interviewcake.com/concept/python/array?>
- Stacks <https://www.interviewcake.com/concept/python/stack?>
- Queues <https://www.interviewcake.com/concept/python/queue?>
- Graphs <https://www.interviewcake.com/concept/python3/graph>
- Book - *Introduction to Programming* - Alan de Freitas (available at <http://www.decom.ufop.br/alan/bcc702/livrocpp.pdf>)

2 Practical Activities

1. Implement the *BubbleSort* algorithm (See section 3.1)
2. Implement the *Brute Force String Matching* algorithm (See Section 3.2).
3. Implement a brute force algorithm to solve the Closest-Pair problem (See section 3.3).
4. Implement a brute force algorithm to solve the problem of finding the convex hull (See section 3.3)

For each implementation, present the complexity analysis for the worst-case and best-case (if applicable) runtime of the algorithm. This analysis should contain:

- A mathematical expression that defines the number of operations (recurrence relation for recursive algorithms or summations for iterative ones)
- Final expression of the cost function
- Indication of the efficiency class (O or Θ). The indication of the class must be justified. You can prove it by definition, by limit, or use results demonstrated in the first exercise list (related to Chapter 2 of the book).