# Python Sorting Algorithms Assignment

Mustafa AbdulRazek

25 Aug 2024

## Task 1: Bubble Sort Implementation

**Objective**: Learn how to implement the Bubble Sort algorithm and understand how sorting works in a step-by-step manner.

## Task Description

Bubble Sort is a simple sorting algorithm that repeatedly steps through the list, compares adjacent elements, and swaps them if they are in the wrong order. The process is repeated until the list is sorted. It is called "Bubble Sort" because smaller elements "bubble" to the top of the list.

## Tutorial

For a visual explanation of Bubble Sort, watch this tutorial on YouTube: Bubble Sort Algorithm Explained

## Task

Write a Python function that implements the Bubble Sort algorithm. The function should take a list of numbers as input and sort it in ascending order. Test your function with a list of at least 10 numbers.

## Example Output

```
Original list: [64, 34, 25, 12, 22, 11, 90]
Sorted list: [11, 12, 22, 25, 34, 64, 90]
```

## Task 2: Merge Sort Implementation

**Objective**: Understand the concept of divide and conquer in sorting, and implement the Merge Sort algorithm.

## Task Description

Merge Sort is a more advanced sorting algorithm that uses the divide and conquer approach. The list is divided into two halves, and each half is recursively sorted. Then, the two sorted halves are merged together to produce the final sorted list. Merge Sort is efficient and has a time complexity of $O(n \log n)$, making it faster than Bubble Sort for larger lists.

## Tutorial

For a visual explanation of Merge Sort, watch this tutorial on YouTube: Merge Sort Algorithm Explained

## Task

Write a Python function that implements the Merge Sort algorithm. The function should take a list of numbers as input and sort it in ascending order. Test your function with a list of at least 10 numbers.

## Example Output

```
Original list: [38, 27, 43, 3, 9, 82, 10]
Sorted list: [3, 9, 10, 27, 38, 43, 82]
```