

Lab 03

Topic

Dynamic array in python

Objective

- To understand the working of Bubble Sort by applying it on different datasets.
- To explore its behavior with modifications and optimizations.
- To analyze the performance of Bubble Sort under different scenarios.

Task (1):

Modify your Bubble Sort program to sort number in **descending order**.

Task (2):

Modify Bubble Sort to **print the array after every pass**, so you can see how elements "bubble up."

Example:

- Pass 1: [...]
- Pass 2: [...]
- ... until sorted.

Task (3):

Extend your program to **count and display**:

- Total number of **comparisons** made.
- Total number of **swaps** performed.

Task (4):

Run Bubble Sort on:

1. A **sorted array**
2. A **reverse-sorted array**
3. A **random array**

Compare and display the **number of comparisons and swaps** for each case.

Task (5):

Implement the optimized Bubble Sort that **stops early if no swaps are made in a pass** (best-case performance)