Task-1:

Augment your Dynamic Safe Array class to have a Bubble Sort function which at any stage can be called to sort the DSA using bubble sort.

Task 2:

If the array is already sorted, we don't want to continue with the comparisons. This can be achieved with a modified bubble sort. Update your DSA class to have a modified bubble sort function which will sort the DSA whenever called.

Task-3:

Insertion sort is an intelligent sorting algorithm, as it only swaps the data if it is not already sorted and works really well when we have partially sorted data. Add the functionality of insertion sort in your DSA class.

Task-4:

Create a Person class which has following attributes:

- First Name
- Last Name
- Birth Year
- Birth Month
- Birth Date

Develop C++ solution such that day month and year are taken as input for *N* persons and perform Sorting based on year, month and day using Selection Sort.

Task 5:

Given an array **arr[ ]** of length **N** consisting cost of **N** toys and an integer **K** the amount with you. The task is to find maximum number of toys you can buy with **K** amount.

Example 1:
Input:
N = 7
K = 50
arr[] = {1, 12, 5, 111, 200, 1000, 10}
Output: 4
Explanation: The costs of the toys
you can buy are 1, 12, 5 and 10.
Example 2:
Input:
N = 3
K = 100
arr[] = {20, 30, 50}
Output: 3
Explanation: You can buy all toys

Task 6:

Create a single class Sort, which will provide the user the option to choose between all 4 sorting techniques. The class should have following capabilities

- Take an array and a string (indicating the user choice for sorting technique) as input and perform the desired sorting.
- Should allow the user to perform analysis on a randomly generated array. The analysis provides number of comparisons and number of swaps performed for each technique.
  - After printing all the results, the class should highlight the best and worst techniques.

Task 7:

Upgrade your Sort class to perform sorting methods on linked lists. Using the concepts of OOP, integrate your sorting class with your linked list class (Your choice, can be Singly/Doubly/Circular) to allow the user to perform desired sorting on their linked list.