### Exercise: Sorting Race   
  
\*\*Topic:\*\* ARRAYS   
\*\*Due Date:\*\* 2025-06-26   
  
\*\*Description:\*\*   
In this exercise, you will implement a sorting race where multiple players compete to sort an array of numbers. The game will utilize two distinct sorting algorithms—Bubble Sort and Quick Sort—and each player will have the opportunity to choose their preferred sorting method. The program will measure the time taken by each algorithm to sort a predefined array, and at the end of the race, display the results, determining which sorting method was the fastest.  
  
\*\*Requirements:\*\*  
  
1. The array will be represented using a dynamic array, allocating memory for it based on user input. You will create struct `SortingRace` that will hold the necessary data for the sorting race:  
 ```c  
 // Structure to represent the sorting race  
 typedef struct {  
 int\* array; // Stores the numbers to be sorted  
 int size; // Size of the array  
 } SortingRace;  
 ```  
  
2. Management of the game:  
 - The user will input the size of the array and the elements of the array.  
 - Players will be prompted to choose a sorting algorithm (Bubble Sort or Quick Sort).  
 - The program will sort the array using the selected algorithm and track the time taken.  
 - The program will display the sorted array along with the sorting time.  
 - Players can repeat the race with different inputs until they choose to exit the game