**Problem Statement:**

You are required to implement a **Sorting and Searching System for Employee Records**. Each employee has the following attributes:

* **ID** (an integer)
* **Name** (a string)
* **Salary** (a double)

The system should allow the following operations:

1. **Sort the employees by salary** using **quick sort**. The sorting must be in **descending order**.
2. **Search for an employee** by their ID using **binary search**.
3. Implement these functionalities using **function pointers** for comparison in the sorting algorithm and **custom comparison logic** for the search.

**Learnings from the Problem Statement:**

1. **Sorting Algorithms:**
   * **Bubble Sort** is inefficient for large datasets, but the program demonstrates the more efficient **quick sort** using the qsort function. This function is versatile and works with any data type using a comparison function.
2. **Function Pointers and Flexibility:**
   * The use of **function pointers** enables custom sorting and searching logic without modifying the core sorting and searching algorithms. This allows flexibility in handling different data types and structures.
3. **Binary Search:**
   * **Binary search** significantly improves search efficiency, especially in sorted arrays. The program demonstrates how bsearch can be customized to search for a specific attribute (like an ID) in a structured dataset.

