🤔 Sorting Questions

1. Inventory Management

> A warehouse manager wants to sort the weights of packages (can be negative if mislabeled). Use a sorting algorithm to identify the 3 lightest and 3 heaviest packages.

Sample Input:

```
arr = {12, -5, 23, 7, -20, 4, 15, -2}
```

Sorted Output:

```
Sorted: {-20, -5, -2, 4, 7, 12, 15, 23}
Lightest 3: -20, -5, -2
Heaviest 3: 12, 15, 23
```

---

2. Temperature Logs

> Sort the temperature data to find the **maximum temperature jump** between any two **consecutive readings**.

Sample Input:

```
arr = {22, 25, 19, 35, 28, 30}
```

Sorted Output:

```
Sorted: {19, 22, 25, 28, 30, 35}
Max jump: 5  // between 30 and 35
```

Note: Sorting helps us detect max jump by comparing consecutive sorted values.

---

3. Leaderboard Ranking

> Scores from a game need to be sorted in **descending** order to assign ranks.

Sample Input:

```
arr = {150, 200, 180, 200, 170}
```

Modified Output (descending):

```
Sorted: {200, 200, 180, 170, 150}
Ranks: 1, 1, 3, 4, 5
```

---

4. Merge Sensor Data

> Merge two **already sorted** arrays from two sensors into one sorted array.

Sample Input:

```
arr1 = {3, 5, 9}
arr2 = {2, 4, 10}
```

Merged Output:

```
Merged: {2, 3, 4, 5, 9, 10}
```

---

5. Error Correction in Logs

> For an array that is **almost sorted**, which algorithm works best?

Sample Input:

```
arr = {10, 20, 30, 25, 40, 50}
```

Sorted Output:

```
Sorted: {10, 20, 25, 30, 40, 50}
```

> Efficient for nearly sorted arrays due to its `O(n)` best case.
