

本地練習 (5 題)

每題寫成一個 `.java` (含 `main`)，用 `./run.sh 你的檔.java` 編譯執行；卡住再看 [參考解答](#)。本練習聚焦 CH19：線性搜尋、二分搜尋、氣泡排序、選擇排序、判斷陣列是否已排序。資料一律用初始化清單給死或在程式內呼叫時傳入，不需鍵盤輸入。

第 2 題 二分搜尋 (已排序陣列找值)

給已排序 `int[] a = {2, 5, 8, 12, 16, 23, 38, 56}`，寫 `static int binarySearch(int[] a, int key)` 用每次砍半找值；在 `main` 找 `23` (印 `Found 23 at index 5`) 與 `7` (印 `7 not found (-1)`)。

1. `low=0`、`high=a.length-1`；`while (low <= high)` 2. `mid=(low+high)/2`；命中回 `mid`、`key`

易錯：條件用 `low < high` 漏一格；邊界沒 ± 1 造成無窮迴圈；陣列沒排序就用二分。參考解答：[參考解答/BinarySearch.java](#)

第 5 題 判斷陣列是否已排序 (遞增)

寫一個 `static boolean isSorted(int[] a)`，判斷陣列是否為「非遞減」（每一格都 `<=` 下一格）；在 `main` 測 `{1, 3, 3, 5, 8}`（印 `true`）與 `{1, 3, 2, 5}`（印 `false`）。

- `for (int i = 0; i < a.length - 1; i++)`：只要發現 `a[i] > a[i+1]` 就 `return false`
- 迴圈跑完都沒違反 \rightarrow `return true`
- `main` 呼叫兩次、分別印出 boolean 結果

易錯：迴圈邊界寫 `i < a.length` 使 `a[i+1]` 越界（要 `length-1`）；把「相等」當未排序（非遞減允許相等，用 `>` 而非 `>=` 判違反）。參考解答：[參考解答/IsSorted.java](#)