|  |
| --- |
| **CHAPTER 3 HOMEWORK** |

**PROBLEM 1:**

**𝐴 = < −9,−9, −5,−2, 0, 3, 7, 7, 10, 15 >**

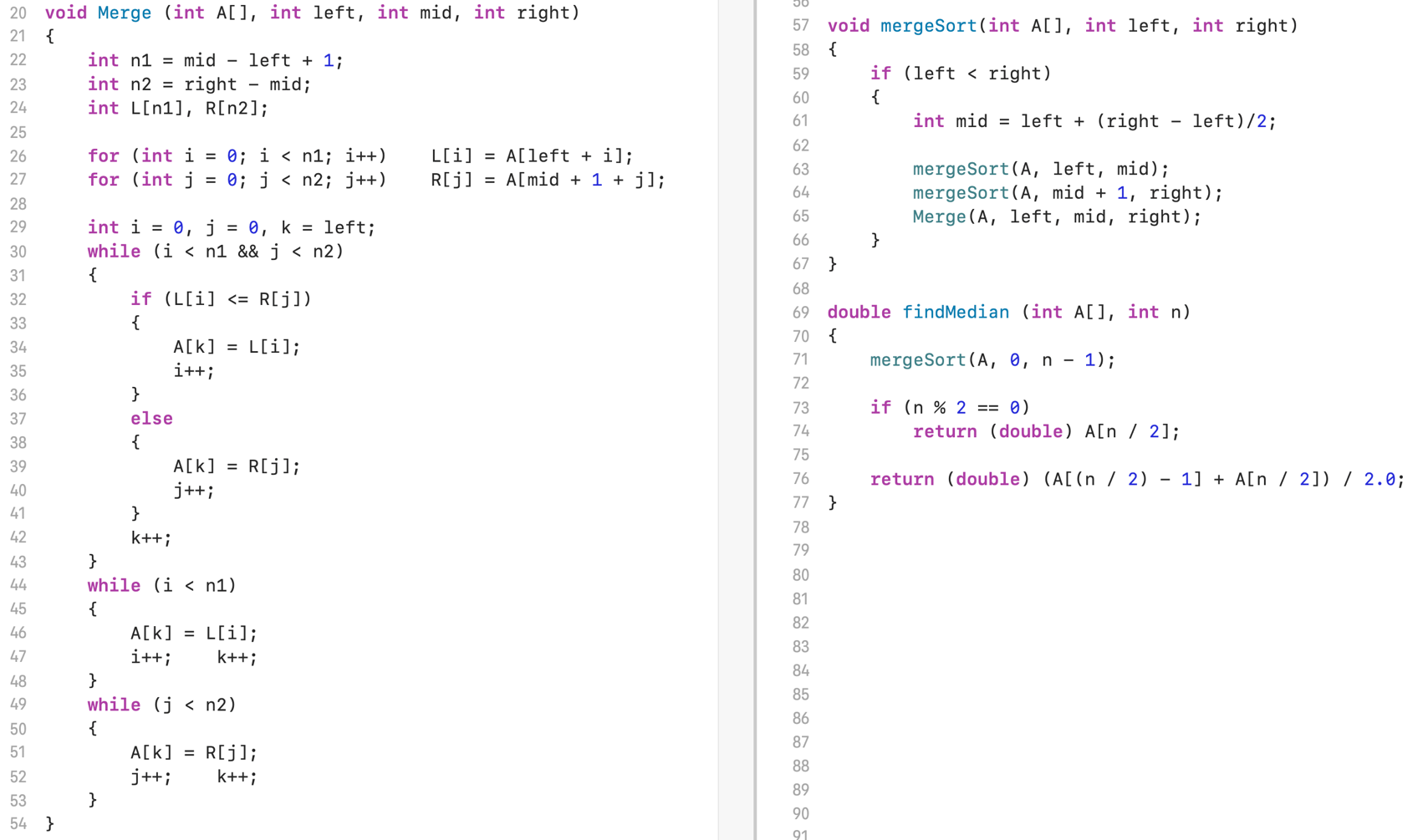
1. Để tìm **𝑥 = −9**, số phép so sánh trong thuật toán

* Linear Search là 1
* Binary Search là 2
* Nhận xét:

x = -9 = a[0] là trường hợp tốt nhất của Linear Search Độ phức tạp: **O (1)**. Sử dụng Linear Search sẽ nhanh hơn và tốt hơn Binary Search (**O (logn)**) trong trường hợp này.

1. Trong trường hợp sử dụng Binary Search, giá trị trả về là 1.

**PROBLEM 2:**

****

Độ phức tạp thuật toán:

**PROBLEM 3:**

Thuật toán tìm kiếm nhị phân được thực hiện trên mảng đã được sắp xếp (VD: mảng được sắp xếp tăng dần)

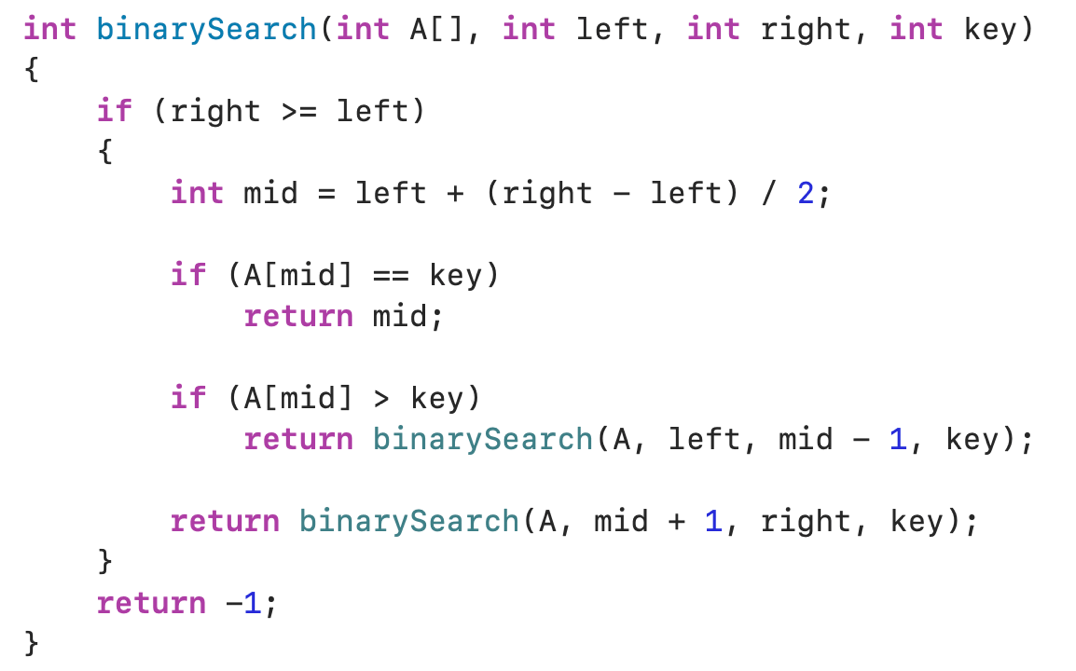
* **Ý tưởng**

***Bước 1:***

* Gọi left là vị trí đầu tiên của mảng (0)
* Gọi right là vị trí cuối cùng của mảng (n - 1)
* Gọi mid là vị trí ở giữa mảng (mid = left + (right – left) / 2)

***Bước 2:*** So sánh

* Nếu key cần tìm bằng với giá trị tại vị trí mid, trả về mid
* Nếu key < giá trị tại vị trí mid, thực hiện tìm kiếm key trên khoảng []
* Nếu key > giá trị tại vị trí mid, thực hiện tìm kiếm key trên khoảng []
* **Thuật toán:**



Độ phức tạp thuật toán:

**PROBLEM 4:**

**𝐴 = < 4, 8, 12, 14, 20, 24 >**

1. **key = 2**

* Ta có: key < A[mid] (2 < 12)
* Thực hiện tìm kiếm key trên khoảng []

Những phần tử được so sánh với key trong mảng A là 4, 8

1. **key = 8**

* Ta có: key < A[mid] (8 < 12)
* Thực hiện tìm kiếm key trên khoảng []

Những phần tử được so sánh với key trong mảng A là 4, 8

1. **key = 15**

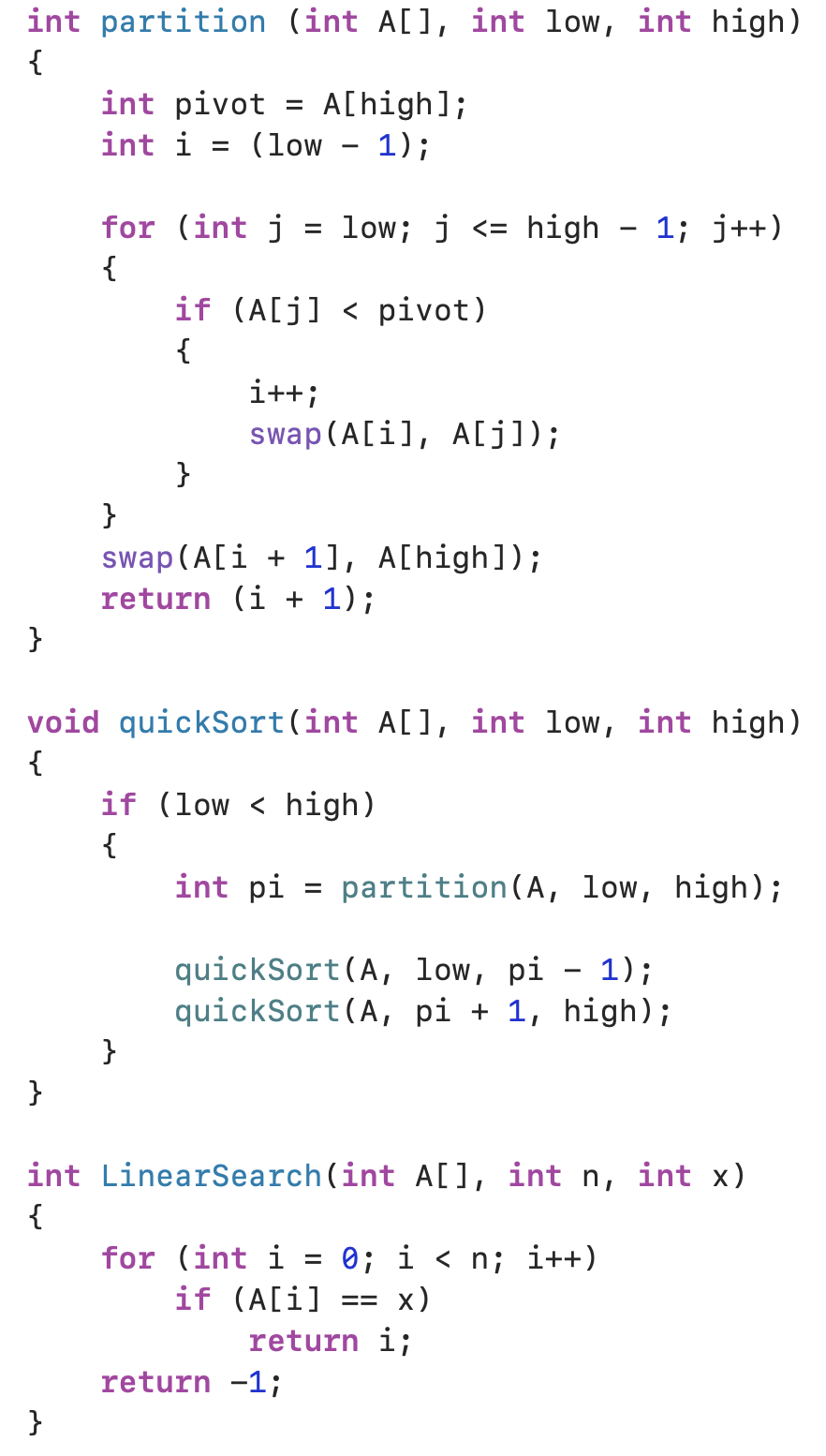
* Ta có: key > A[mid] (15 > 12)

Thực hiện tìm kiếm key trên khoảng []

Những phần tử được so sánh với key trong mảng A là 14, 20, 24

**PROBLEM 5:**

**QUICK SORT**

****

1. Tìm kiếm tuần tự trên mảng đã được sắp xếp không tốt hơn mảng chưa được sắp xếp vì độ phức tạp thuật toán của cả 2 vẫn là **O (n)**.
2. Thuật toán này không nhanh hơn thuật toán Binary Search.

Vì độ phức tạp thuật toán khi sort là như nhau (sử dụng chung thuật toán sort), nhưng độ phức tạp thuật toán của Binary Search < độ phức tạp thuật toán của Linear Search ( **O(logn) < O (n)** )

