

BÀI TẬP VỀ NHÀ – CS112.M21.KHCL

NHÓM 13

Phần I: TỰ LUẬN

Câu 1: Cho thuật toán tìm kiếm tuần tự như bên dưới. Xác định độ phức tạp của thuật toán này

```
int Search(int x, int a[] , int n){
    int i;
    int found;
    i = 0;
    while (i <= n - 1 && ! found)
        if (x == a[i])
            found = 1;
        else i++;
    return found;
}
```

- Các phép gán và khai báo ở bên ngoài vòng lặp while có độ phức tạp là $O(1)$
- Các phép gán và so sánh trong vòng lặp while có độ phức tạp là $O(1)$ (Quy tắc cộng lấy max)
- Vòng lặp while sẽ lặp n lần nên có độ phức tạp là $O(n * 1) = O(n)$ (Quy tắc nhân)
- Vậy độ phức tạp của thuật toán tìm kiếm tuần tự là $\max(O(1), O(n)) = O(n)$ (Quy tắc cộng lấy max)

Câu 2: Cho mảng A có n phần tử số nguyên (với $n > 1$). Bài toán yêu cầu tìm hiệu lớn nhất khi lấy bất kỳ $A[x] - A[y]$ với $x > y$.

Ví dụ:

Mảng A = [9, 1, 2, 8], n = 4

$\text{maxDiff}(A, n) = 7$ // bởi vì $A[3] - A[1] = 8 - 1 = 7$

```
def maxDiff(a, n):
    min_value, pos, maxdiff = a[0], 0, 0
    for i in range(1, n):
        if a[i] < min_value:
            min_value = a[i]
            pos = i

        if i > pos:
            diff = a[i] - min_value
            if diff > maxdiff:
                maxdiff = diff
    return maxdiff
```

Thuật toán có độ phức tạp $O(n)$

PHẦN II: TRẮC NGHIỆM

- 1.C
- 2.B
- 3.A
- 4.D
- 5.D