

# Cấu trúc dữ liệu và thuật toán

## Giải thuật sắp xếp đếm (CountingSort)

### 1 Bài 1

Cho cài đặt của giải thuật sắp xếp CountingSort dãy  $a[0..n-1]$  theo thứ tự tăng dần như sau:

Đầu vào: Mảng  $a$  có  $n$  phần tử cần sắp xếp,  $k$  là giá trị của phần tử lớn nhất

Mảng phụ  $b$  có kích thước  $k$

Đầu ra: Mảng  $c$  đã được sắp xếp

```
void countSort(int a[], int b[], int c[], int k, int n) {
```

```
    // k: Giá trị phần tử lớn nhất
```

```
    // Đếm:  $b[i]$  số phần tử có giá trị  $i$ 
```

```
    for (int i = 0; i <= k; i++) // (A)
```

```
         $b[i] = 0;$ 
```

```
    for (int i = 0; i < n; i++) // (B)
```

```
         $b[a[i]]++;$ 
```

```
    // Chinh:  $b[i] =$  số phần tử  $\leq i$ 
```

```
    for (int i = 1; i <= k; i++) // (C)
```

```
         $b[i] += b[i - 1];$ 
```

```
    // Sắp xếp
```

```
    for (int i = n - 1; i >= 0; i--) { // (D)
```

```
         $c[b[a[i]] - 1] = a[i];$ 
```

```
         $b[a[i]] = b[a[i]] - 1;$ 
```

```
    }
```

```
}
```

Thực hiện các bước của countSort() với mảng  $a[6] = \{10, 22, 55, 11, 88, 77\}$

- Kết quả của mảng  $b$  sau vòng lặp for tại vị trí (A)
- Kết quả của mảng  $b$  sau vòng lặp for tại vị trí (B)
- Kết quả của mảng  $b$  sau vòng lặp for tại vị trí (C)
- Kết quả của mảng  $c$  và  $b$  sau mỗi giá trị của  $i$  trong vòng lặp for tại vị trí (D)