

# Bài tập thực hành

## Cấu trúc dữ liệu và thuật toán

### Sắp xếp đếm (A)

#### 1. Bài 1

Viết chương trình bằng ngôn ngữ lập trình C để hoàn thành giải thuật sắp xếp đếm.

Cho mảng k: k[0], k[2], ..., k[n-1] chứa dãy n số nguyên.

Cho khai báo của hàm sắp xếp dãy n số theo thứ tự tăng dần dùng giải thuật sắp xếp đếm như sau:

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

trong đó a[] là mảng cần sắp xếp, c[] là mảng kết quả, k là giá trị của phần tử lớn nhất, n là số lượng phần tử của mảng a, b[] là mảng phụ có kích thước k.

Cho khai báo của hàm in ra màn hình dãy n phần tử trong mảng k[0..n-1]:

```
void print_array(int k[], int n);
```

a) Hoàn thiện mã chương trình của hàm countSort()

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

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

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

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

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

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

```
        b[i] += ____ ;
```

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

```
        c[ ____ ] = a[i];
```

```
        b[a[i]] = ____ ;
```

```
}
```

```
}
```

b) Hoàn thiện mã chương trình của hàm `print_array()`

c) Hoàn thiện mã chương trình của hàm `main()`

```
int main(void) {  
    int a[10] = { 42, 23, 74, 11, 65, 58, 94, 36, 99, 87 };  
    int c[10];  
    int n = 10;  
    int amax;  
    int * b;  
  
    for (int i = 0; i < n; i++)  
        if (a[i] > amax) amax = ____ ;  
  
    b = (int *) malloc( ____ );  
    countSort(a, b, c, amax, n);  
  
    printf("\nMang truoc khi sap xep: ");  
    print_array(a, n);  
    printf("\nMang sau khi sap xep: ");  
    print_array(c, n);  
  
    free(b);  
  
    return 0;  
}
```

## **2. Bài 2**

Hoàn thành bài 1 với sắp xếp giảm dần.