

Chương 1

THUẬT TOÁN QUICK SORT

1. BÀI TOÁN DẪN NHẬP

- **Bài toán:** Cho mảng một chiều các số nguyên và giá trị x . Hãy tách mảng a ban đầu thành 2 mảng b và c sao cho mảng b chỉ chứa các giá trị nhỏ hơn x , mảng c chứa các giá trị lớn hơn x .

1. BÀI TOÁN DẪN NHẬP

```
11. void Split(int a[], int n,  
               int x,  
               int b[], int &k,  
               int c[], int &l)  
12. {  
13.     k = l = 0;  
14.     for(int i=0; i<n; i++)  
15.         if(a[i]<x)  
16.             b[k++] = a[i];  
17.         else  
18.             if(a[i]>x)  
19.                 c[l++] = a[i];  
20. }
```

2. TƯ TƯỞNG THUẬT TOÁN QUICK SORT

- Thuật toán quick sort chia không gian cần sắp xếp thành 2 không gian con là không gian con 1 và không gian con 2. Không gian con 1 là không gian mà tất cả các phần tử thuộc không gian này đều nhỏ hơn tất cả các phần tử thuộc không gian con 2.
 - + Nếu không gian con thứ nhất có nhiều hơn một phần tử thì sắp xếp không gian con này bằng thuật toán Quick Sort.
 - + Nếu không gian con thứ hai có nhiều hơn một phần tử thì sắp xếp không gian con này bằng thuật toán Quick Sort.

3. HÀM CÀI ĐẶT

```
10. void QuickSort(int a[], int n)
11. {
12.     if (n <= 1)
13.         return;
14.     int b[100]; int k;
15.     int c[100]; int l;
16.     int TrongTai = a[0];
17.     Split(a, n, TrongTai, b, k, c, l);
18.     QuickSort(b, k);
19.     QuickSort(c, l);
20.     for(int i=0; i<k; i++)
21.         a[i] = b[i];
22.     for(int i=0; i<n-k-l; i++)
23.         a[k+i] = TrongTai;
24.     for(int i=0; i<l; i++)
25.         a[k+(n-k-l)+i] = c[i];
26. }
```

3. HÀM CÀI ĐẶT

```
11. void Split(int a[], int n,  
               int x,  
               int b[], int &k,  
               int c[], int &l)  
12. {  
13.     k = l = 0;  
14.     for(int i=0; i<n; i++)  
15.         if(a[i]<x)  
16.             b[k++] = a[i];  
17.         else  
18.             if(a[i]>x)  
19.                 c[l++] = a[i];  
20. }
```