

Bài 2.9: Thuật toán sinh xâu nhị phân

- ✓ Mô tả bài toán
- ✓ Thuật toán tổng quát
- ✓ Ví dụ minh họa
- ✓ Bài tập thực hành

Mô tả thuật toán

- Thuật toán sinh xâu nhị phân độ dài n áp dụng thuật toán sinh cấu hình kế tiếp để giải quyết vấn đề.
- Bài toán: cho số nguyên dương n . Viết chương trình sinh toàn bộ các cấu hình của xâu nhị phân độ dài n theo thứ tự từ điển.
- Xâu $B = \{b_1, b_2, \dots, b_n\}$ với $b_i \in \{0, 1\}$ và $i \in \{1, n\}$ là xâu nhị phân độ dài n .
- Mỗi cấu hình của xâu là bộ các giá trị gồm n bit có thứ tự khác nhau.
- Với một số nguyên dương n cho trước, ta có 2^n xâu nhị phân thỏa mãn.

Thuật toán tổng quát

➤ Sau đây là mã giả thuật toán sinh xâu nhị phân độ dài n:

```
// thuật toán sinh xâu nhị phân kế tiếp độ dài n
bool nextBinaryString(arr[], n) {
    i = n - 1;           // bắt đầu từ phần tử cuối xâu
    while(i >= 0 && arr[i] != 0) { // tìm phần tử đầu tiên bằng 0
        arr[i] = 0;      // gán phần tử tại vị trí i = 0
        i--;             // dịch sang vị trí bên trái
    }
    if(i >= 0) {          // nếu cấu hình đang xét chưa phải cấu hình cuối
        arr[i] = 1;       // gán phần tử tại vị trí tìm được = 1
        return false;     // trả về thông báo cấu hình chưa phải cuối cùng
    } else {              // ngược lại, đây là cấu hình cuối
        return true;      // trả về thông báo đây là cấu hình cuối
    }
}

// thuật toán sinh tất cả các xâu nhị phân n phần tử
void generate(arr[], n) { // arr: mảng chứa xâu nhị phân độ dài n
    isFinal = false;     // biến đánh dấu đã đến cấu hình cuối cùng
    while (!isFinal)     // nếu chưa phải cấu hình cuối cùng
    {
        output(arr, n);  // hiển thị mảng arr
        isFinal = nextBinaryString(arr); // sinh xâu kế tiếp
    }
}
```

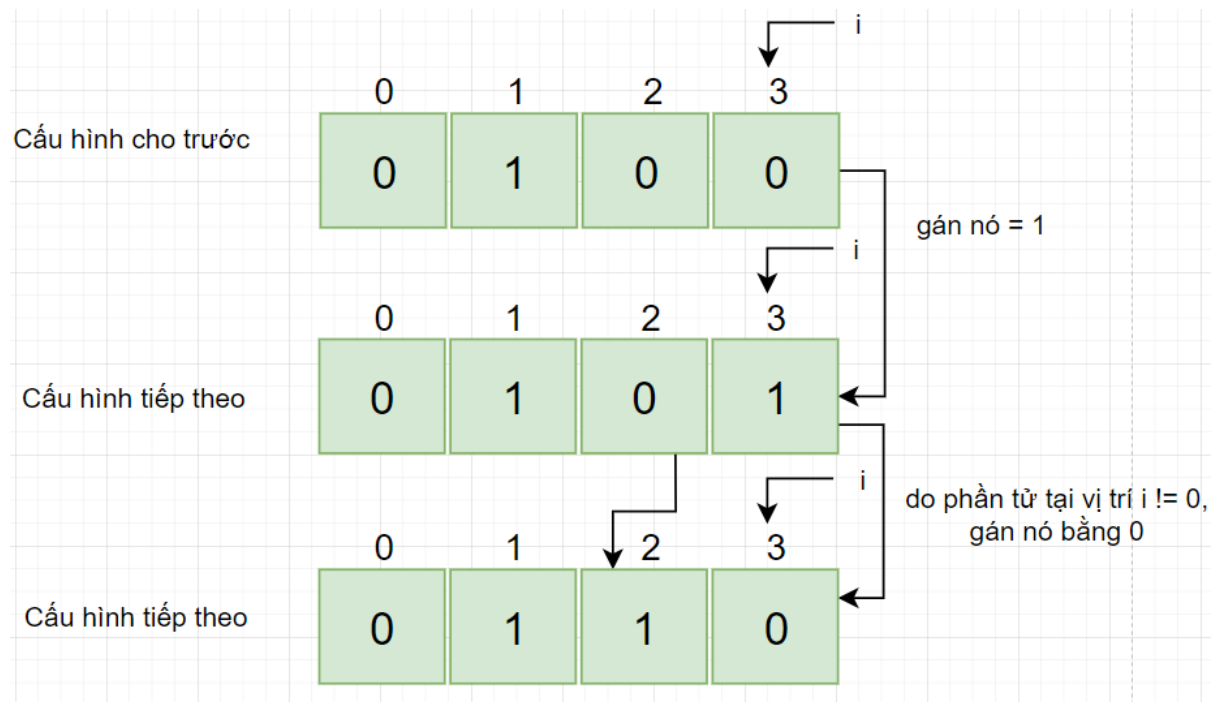
Thuật toán tổng quát

➤ Sau đây là mã thật thuật toán sinh xâu nhị phân độ dài n:

```
bool nextBinaryString(int* arr, int n) {  
    int i = n - 1;  
    while (i >= 0 && arr[i] != 0) {  
        arr[i] = 0;  
        i--;  
    }  
    if (i >= 0) {  
        arr[i] = 1;  
        return false;  
    }  
    else {  
        return true;  
    }  
}  
  
void generate(int* arr, int n) {  
    bool isFinal = false;  
    while (!isFinal) {  
        output(arr, n);  
        isFinal = nextBinaryString(arr, n);  
    }  
}
```

Ví dụ minh họa

➤ Ví dụ cho một cấu hình 0100, tìm cấu hình kế tiếp:



Nội dung tiếp theo

Thuật toán sinh hoán vị chính tắc