

Bài toán liệt kê

TOÁN RỜI RẠC

NỘI DUNG

- ✘ Giới thiệu
- ✘ Phương pháp sinh
- ✘ Thuật toán quay lui

GIỚI THIỆU

BÀI TOÁN LIỆT KÊ

- ✘ *Khái niệm: đưa ra danh sách tất cả cấu hình tổ hợp có thể có, thỏa mãn những điều kiện cho trước*
- ✘ *Yêu cầu:*
 - + *Đầy đủ*
 - + *Không trùng lặp*

PHƯƠNG PHÁP SINH

Từ bộ dữ liệu bắt đầu, sinh ra bộ dữ liệu kế tiếp cho đến khi kết thúc

PHƯƠNG PHÁP SINH

✘ Điều kiện:

- + Có thể xác định được 1 thứ tự trên tất cả các tổ hợp thỏa mãn điều kiện.
- + Xây dựng được thuật toán để xác định ra tổ hợp liền sau một tổ hợp được xét

✘ Thuật toán:

- + Bước 1: xác định tổ hợp đầu tiên
- + Bước 2: xây dựng tổ hợp kế tiếp của tổ hợp mới nhất
- + Bước 3: kiểm tra xem đã là tổ hợp cuối cùng chưa?
 - ✘ Nếu chưa thì quay lại bước 2
 - ✘ Nếu là tổ hợp cuối cùng rồi thì kết thúc

VÍ DỤ

- ✘ Liệt kê tất cả các xâu nhị phân 4 kí tự
- ✘ Quy mỗi xâu \Rightarrow 1 số nhị phân

\Rightarrow số nhỏ nhất: 0000; số lớn nhất: 1111

Nếu ta lấy 0000 là tổ hợp bắt đầu, 1111 là tổ hợp kết thúc

Cho trước 1 số: xxxx

Thì tổ hợp liền sau sẽ là $xxxx + 1$

(hoặc theo chiều ngược lại)

VÍ DỤ

- ✘ Liệt kê tất cả các xâu nhị phân 4 kí tự

0000	1000
0001
0010
0011
0100
0101
0110
0111	1111

VÍ DỤ

Liệt kê tất cả các tập con của 1 tập n phần tử.

(cụ thể: $n = 10$, các phần tử: A, B, C, D, ... I, K)

Coi 1 tập con tương ứng với 1 xâu nhị phân

Trong đó:

- Vị trí thứ $i = 1$ là có phần tử thứ i
- Vị trí thứ $i = 0$ là không có phần tử thứ i

Sử dụng phương pháp sinh để tạo ra xâu nhị phân n phần tử rồi từ xâu \Rightarrow tập tương ứng.

VÍ DỤ

Cho trước số tự nhiên $n \geq 2$, tìm cách liệt kê tất cả các trường hợp có thể tách n thành tổng các số tự nhiên khác không nhỏ hơn nó.

$$6 = 5 + 1$$

$$= 4 + 2 = 4 + 1 + 1$$

$$= 3 + 3 = 3 + 2 + 1 = 3 + 1 + 1 + 1$$

$$= 2 + 2 + 2 = 2 + 2 + 1 + 1 = 2 + 1 + 1 + 1 + 1$$

$$= 1 + 1 + 1 + 1 + 1 + 1$$

Tổng số: 10 trường hợp

VÍ DỤ

Cho trước số tự nhiên $n \geq 2$, tìm cách liệt kê tất cả các trường hợp có thể tách n thành tổng các số tự nhiên khác không nhỏ hơn nó.

Cách chia: $n = a_1 + a_2 + \dots$

Trong đó $a_1 \geq a_2 \geq \dots$

Các tổ hợp sắp theo chiều giảm dần của a_1 , rồi đến a_2, \dots

⇒ Tổng đầu tiên có 2 phần tử $(n-1, 1)$

Tổng cuối cùng có n phần tử $(1, 1, \dots, 1)$

THUẬT TOÁN QUAY LUI (BACKTRACKING)

VẤN ĐỀ CỦA PHƯƠNG PHÁP SINH

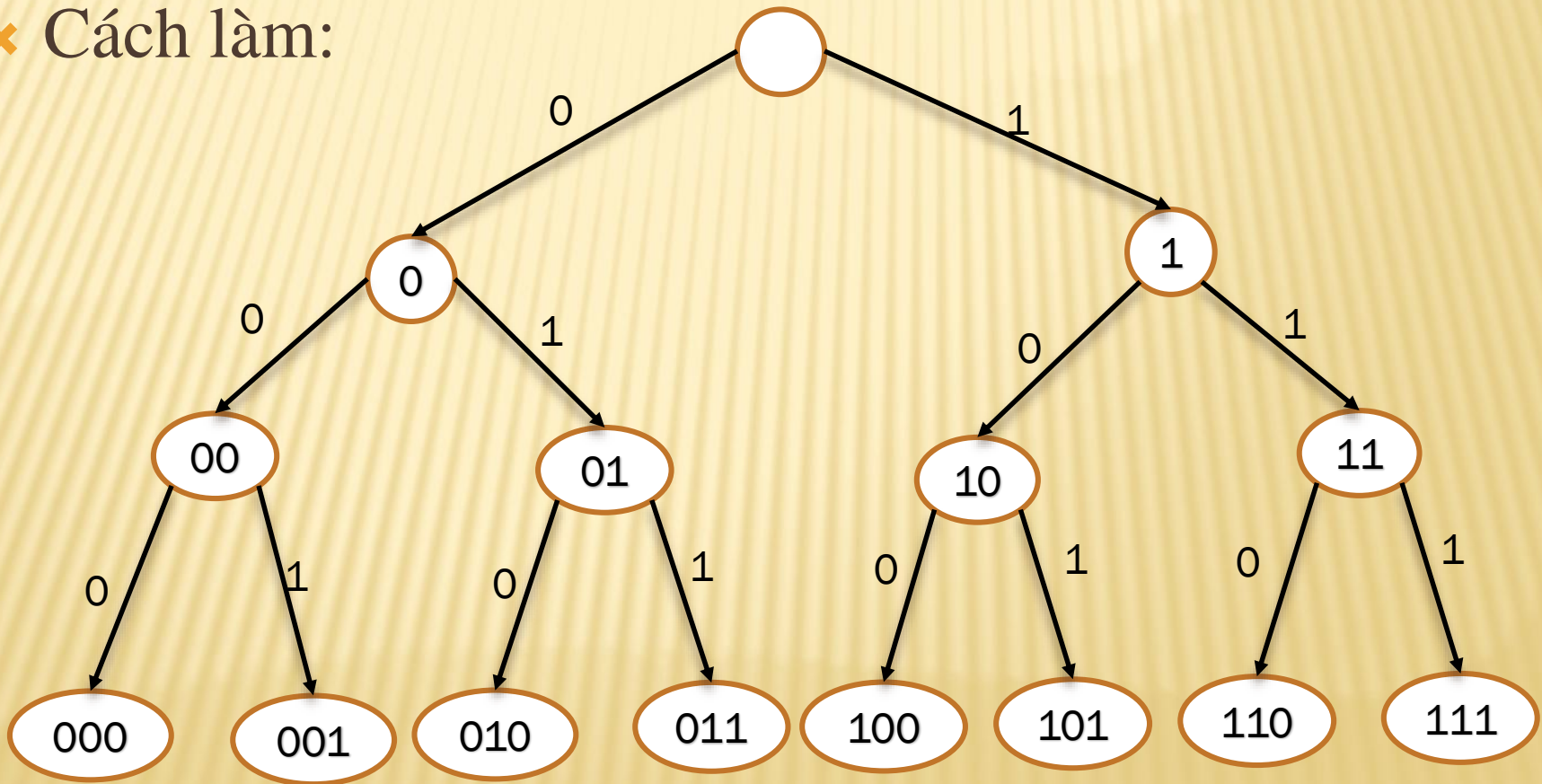
- ✘ Cách sinh ra cấu hình tiếp theo
- ✘ Xác định cấu hình bắt đầu và kết thúc

VÍ DỤ

- ✘ Liệt kê các xâu nhị phân có độ dài n
- ✘ Cách làm:
 - + try(i): Chọn giá trị cho xâu bộ phận đang có độ dài
 - + **Tập giá trị** của từng phần tử: $D = \{0, 1\}$
 - + Thành công: Khi chọn được giá trị cho thành phần thứ $i = n$
 - + Thông báo kết quả: Hiển thị các dãy nhị phân độ dài n

VÍ DỤ

- ✘ Liệt kê các xâu nhị phân có độ dài n
- ✘ Cách làm:



VÍ DỤ

- ✘ Liệt kê các hoán vị của các giá trị từ 1 đến n
- ✘ Cách làm:
 - + try(i): chọn giá trị cho dãy hiện đang có độ dài I
 - + Tập giá trị của từng phần tử: $D = \{1, 2, \dots, n\}$
 - + Giá trị có thể chọn: Các giá trị j chưa được chọn trước đó
 - + Chọn giá trị: Chọn j và đánh dấu j đã chọn cho x_i
 - + Thành công: Khi chọn được thành phần thứ $i = n$
 - + Thông báo kết quả: Hiển thị n số của hoán vị
 - + Hủy chọn: Đánh dấu j chưa được chọn

VÍ DỤ

```
void try(int i) {  
  
    for (int j = 1; j <= n; j++)  
        if (mark[j] == false) {  
            x[i] = j;  
            mark[j] = true;  
            if (i == n)  
                print_result();  
            else try(i + 1);  
            mark[j] = false;  
        }  
}
```

Ý TƯỞNG

- ✘ Đáp án là một chuỗi có thứ tự (a_1, a_2, \dots, a_n) ; mỗi a_i có một số khả năng để chọn lựa.
- ✘ Đáp án = đường đi từ a_1 đến a_n . Qua mỗi điểm ta chọn 1 giá trị cho vị trí tương ứng
- ✘ Nếu chọn đúng n giá trị \Rightarrow được 1 đáp án.
- ✘ Nếu đi đến giữa chừng mà k đi được nữa thì ta quay lại điểm cần chọn lựa trước đó để chọn một giá trị khác

Ý TƯỞNG

- ✘ Tổ hợp cần tìm là một bộ có dạng (a_1, \dots, a_n)
- ✘ Tại mỗi bước
 - + Giả sử: đã xác định được *lời giải bộ phận* $a = (a_1, \dots, a_{i-1})$
 - + Mở rộng a bằng cách thêm một giá trị vào sau bộ hiện tại.
 - + Kiểm tra xem đã thu được đáp án hoàn chỉnh chưa?
 - ✘ Nếu chưa: KT xem còn tiếp tục mở rộng được không? Nếu được thì ta tiếp tục, lặp lại quy trình trên
 - + Nếu đã thu được 1 đáp án hoặc không mở rộng được tiếp thì (in ra đáp án hoàn chỉnh và) xóa phần tử cuối cùng trong a và thử một giá trị khác cho vị trí đó nếu vẫn còn khả năng và lặp lại quy trình

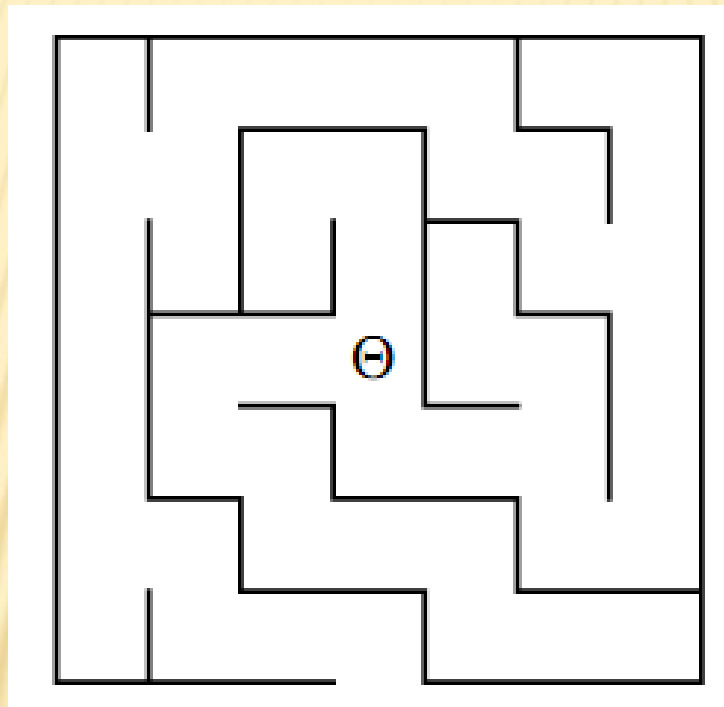
THUẬT TOÁN QUAY LUI

```
Try(int i){  
    for (mọi giá trị có thể chọn){  
        <Thử cho  $a_i = V$ >;  
        if ( là phần tử cuối cùng trong cấu hình) then  
            <Thông báo bộ giá trị tìm được>  
        else{  
            <Ghi việc nhận giá trị V (Nếu cần)>;  
            Try(i + 1); {Gọi đệ qui để chọn tiếp }  
        }  
        <Nếu cần, bỏ việc thử = V, thử giá trị khác>;  
    }  
}
```

(Thuật toán quay lui sẽ bắt đầu bằng lời gọi Try(1);)

VÍ DỤ

- ✘ Tìm đường đi ra khỏi mê cung:



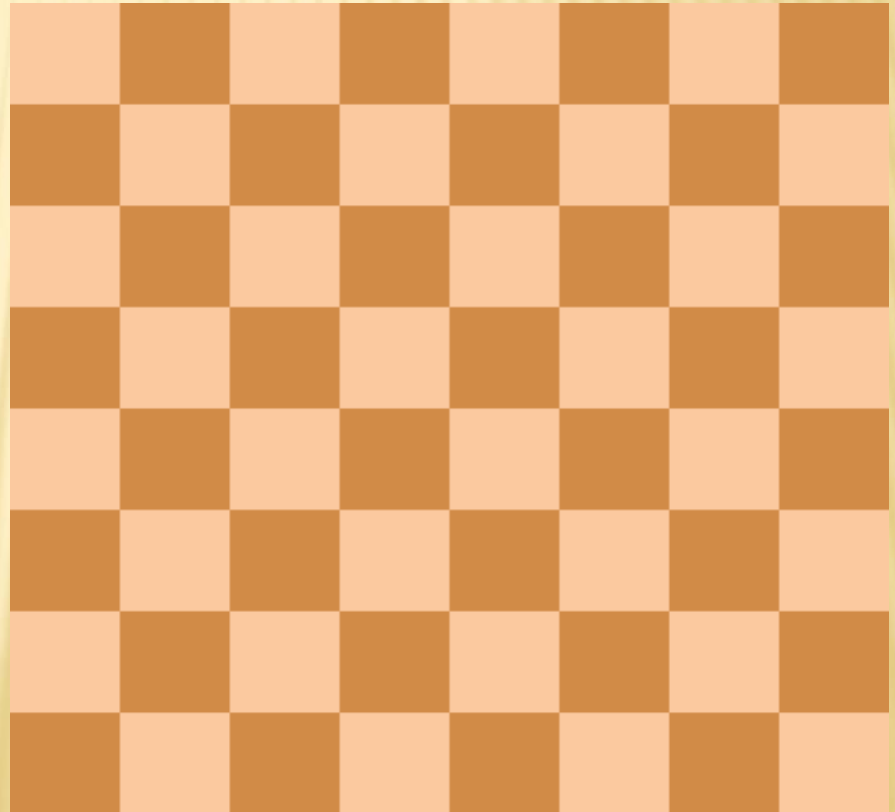
VÍ DỤ

a) Cho 1 bàn cờ 4×4

Tìm cách đặt 4 quân hậu trên bàn cờ để không có 2 quân nào ăn được nhau?

b) Cho bàn cờ 5×5

c) Bàn cờ $n \times n$



VÍ DỤ

Dark Brown			



Dark Brown	Yellow	Yellow	Yellow
Yellow	Yellow		
Yellow		Yellow	
Yellow			Yellow



Dark Brown	Yellow	Yellow	Yellow
Yellow	Yellow	Dark Brown	
Yellow		Yellow	
Yellow			Yellow



Dark Brown	Yellow	Yellow	Yellow
Yellow	Yellow	Dark Brown	Yellow
Yellow	Yellow	Yellow	Yellow
Yellow		Yellow	Yellow

	Dark Brown		



Yellow	Dark Brown	Yellow	Yellow
Yellow	Yellow	Yellow	
	Yellow		Yellow
	Yellow		



Yellow	Dark Brown	Yellow	Yellow
Yellow	Yellow	Yellow	Dark Brown
	Yellow	Yellow	Yellow
	Yellow		Yellow



Yellow	Dark Brown	Yellow	Yellow
Yellow	Yellow	Yellow	Dark Brown
Dark Brown	Yellow	Yellow	Yellow
	Yellow		Yellow



Yellow	Dark Brown	Yellow	Yellow
Yellow	Yellow	Yellow	Dark Brown
Dark Brown	Yellow	Yellow	Yellow
Yellow	Yellow	Dark Brown	Yellow

VÍ DỤ

- ✘ Trò chơi sudoku
- ✘ Tìm ra lời giải cho bảng số?

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9