

Lớp: CS112.P11.CTTN

Nhóm: 14

Sinh viên: Lê Nguyễn Anh Khoa. MSSV: 23520742

Sinh viên: Cáp Kim Hải Anh. MSSV: 23520036

BÀI TẬP

Vận dụng thiết kế thuật toán Lý thuyết trò chơi

I. Bài 1: Đối kháng

1. Đề bài:

Hằng năm ở thành phố X, sẽ tổ chức một cuộc thi đối kháng hai người. Ban đầu người ta sẽ giao cho 2 bạn một số nguyên dương p . Hai người thi đấu theo lượt, A đi trước B. Nếu ai làm cho p bằng 0 thì người đó thắng. Trong một lượt chơi, người chơi thực hiện thao tác sau:

- Nếu p lẻ, người chơi được chọn tăng p hoặc giảm p 1 đơn vị.
- Nếu p chẵn, thì người chơi bắt buộc giảm p xuống 1 nửa $p := p/2$

Cho trước số nguyên dương p . Bạn A luôn đi trước B, nếu cả 2 đều chơi tối ưu thì bạn A luôn thắng được không? (xuất ra màn hình YES nếu A luôn thắng hoặc ngược lại B luôn thắng)

- **Loại trò chơi:** tổng bằng 0. Vì có trạng thái thắng, thua và tổng lợi ích luôn bằng 0
- **Loại biểu diễn:** cây

2. Yêu cầu:

a. $p \leq 10$

- **Ý tưởng:** Sử dụng thuật toán quay lui để duyệt qua tất cả các trường hợp.
 - Nếu tồn tại một nước đi khiến đối thủ rơi vào trạng thái thua, thì trạng thái đó là thắng
 - Nếu mọi nước đi đều dẫn đến trạng thái thắng của đối thủ, như vậy trạng thái đó là thua
- **Mã giả:**

```
def cal(p):
    if p == 0:
        return False
    if p&1:
        return !cal(p-1) | !cal(p+1)
    else:
        return !cal(p/2)
```

- Độ phức tạp:

- Thời gian: $O(2^n)$
- Không gian: $O(1)$

b. $p \leq 10^6$:

- Ý tưởng:

- Sử dụng thuật toán quy hoạch động
- $Dp[i]$: Kết quả nếu đến lượt $p = i$ đi.
- $Dp[i] = 1$ nếu tồn tại cách đi để thắng
- $Dp[i] = 0$ nếu không tồn tại cách đi để thắng
- Nếu p chẵn: $Dp[i] = Dp[i/2]$
- Nếu p lẻ: $Dp[i] = !Dp[i-1] | !Dp[i+1]$
- $Dp[1] = 1, Dp[0] = 0$
- Sử dụng quy hoạch động đệ quy để tính

- Mã giả:

```
dp = [-1] * (10^6 + 5)
dp[0] = False
dp[1] = True
def cal(p):
    if dp[p] != -1:
        return dp[p]
    if p&1:
        dp[p] = !cal(p-1) | !cal(p+1)
    else:
        dp[p] = !cal(p/2)
    return dp[p]
```

- Độ phức tạp:

- Thời gian: $O(p)$
- Không gian: $O(1)$

II. Bài 2: Trò chơi đồng xu

1. Đề bài:

Để cạnh tranh sức hút trò chơi ở thành phố X, tại thành phố Y cũng đã tổ chức một cuộc thi đối kháng hai người, A luôn đi trước B. Ban đầu, 2 người chơi được giao một chồng gồm n đồng xu. Trong lượt chơi, bạn được phép chọn 1 hoặc 2,... hoặc k đồng xu và bốc nó ra khỏi chồng (sau lượt này n sẽ giảm đi x đồng xu, x là số đồng xu bạn bốc ra, $x \leq k$). Nếu không thực hiện bốc được thì bạn thực hiện lượt đó sẽ thua.

Bài toán khá quen thuộc để tăng độ hấp dẫn. Phước sẽ dành cho các bạn câu đố sau. Với những giá trị nào của k ($k \leq n$) mà đảm bảo A luôn thắng. In ra số lượng k thỏa mãn.

- **Loại trò chơi: có tổng bằng 0**

2. Yêu cầu:

a. $n \leq 1000$

- **Ý tưởng:**

- Sử dụng thuật toán quy hoạch động
- $Dp[i]$: Kết quả nếu đến lượt có i đồng xu.
- $Dp[i] = 1$ nếu tồn tại cách đi để thắng
- $Dp[i] = 0$ nếu không tồn tại cách đi để thắng
- $Dp[i] = !Dp[i-1] \mid !Dp[i-2] \mid \dots \mid !Dp[i-k]$
- $Dp[1] = 1, Dp[0] = 0$
- Xuất ra các giá trị k mà $dp[k] = 1$

- **Mã giả:**

```
dp = [False] * (n)
dp[0] = False
dp[1] = True
for i in range(n):
    if dp[i] == False:
        for j in range(i+1, min(n+1, i+k)):
            dp[j] = True

for i in range(1, n+1):
    if dp[i] == True:
        print(i)
```

- **Độ phức tạp:**

- Thời gian: $O(n^2)$
- Không gian: $O(n)$