

## Bài 1 :

**Bước 1:** Gọi hàm `sum_of_numbers(7)`

Kiểm tra điều kiện : `7 == 1` sai nên chuyển xuống thực hiện câu lệnh `else : return 7 + sum_of_numbers(6)`

**Bước 2:** Gọi hàm `sum_of_numbers(6)`

Kiểm tra điều kiện : `6 == 1` sai nên chuyển xuống thực hiện câu lệnh `else : return 6 + sum_of_numbers(5)`

**Bước 3:** Gọi hàm `sum_of_numbers(5)`

Kiểm tra điều kiện : `5 == 1` sai nên chuyển xuống thực hiện câu lệnh `else : return 5 + sum_of_numbers(4)`

Gọi hàm `sum_of_numbers(4)`

**Bước 4:**

Kiểm tra điều kiện : `4 == 1` sai nên chuyển xuống thực hiện câu lệnh `else : return 4 + sum_of_numbers(3)`

**Bước 5:** Gọi hàm `sum_of_numbers(3)`

Kiểm tra điều kiện : `3 == 1` sai nên chuyển xuống thực hiện câu lệnh `else : return 3 + sum_of_numbers(2)`

**Bước 6:** Gọi hàm `sum_of_numbers(2)`

Kiểm tra điều kiện : `2 == 1` sai nên chuyển xuống thực hiện câu lệnh `else : return 2 + sum_of_numbers(1)`

**Bước 7:** Gọi hàm `sum_of_numbers(1)`

Kiểm tra điều kiện :  $1 == 1$  đúng nên hàm trả về giá trị là 1

**Thực hiện lần lượt từ bước 7 lên bước 1 ta được :**

sum\_of\_numbers(1) trả về 1 -> sum\_of\_numbers(2) trả về  $2 + 1 = 3$  -> sum\_of\_numbers(3) trả về  $3 + 3 = 6$  -> sum\_of\_numbers(4) trả về  $4 + 6 = 10$  -> sum\_of\_numbers(5) trả về  $5 + 10 = 15$  -> sum\_of\_numbers(6) trả về  $6 + 15 = 21$  -> sum\_of\_numbers(7) trả về  $7 + 21 = 28$  -> kết thúc chương trình

**Vậy kết quả của chương trình là : 28**

Lê Khắc Dương

## **Bài 2:**

**Bước 1:** Gọi hàm fibonacci(8)

Xét điều kiện  $8 \leq 1$  sai , thực hiện điều kiện của else :  $\text{return fibonacci}(7) + \text{fibonacci}(6)$

**Bước 2:** Gọi hàm fibonacci(7)

Xét điều kiện  $7 \leq 1$  sai , thực hiện điều kiện của else :  $\text{return fibonacci}(6) + \text{fibonacci}(5)$

**Bước 3:** Gọi hàm fibonacci(6)

Xét điều kiện  $6 \leq 1$  sai , thực hiện điều kiện của else :  $\text{return fibonacci}(5) + \text{fibonacci}(4)$

**Bước 4:** Gọi hàm fibonacci(5)

Xét điều kiện  $5 \leq 1$  sai , thực hiện điều kiện của else : `return fibonacci(4) + fibonacci(3)`

**Bước 5:** Gọi hàm `fibonacci(4)`

Xét điều kiện  $4 \leq 1$  sai , thực hiện điều kiện của else : `return fibonacci(3) + fibonacci(2)`

**Bước 6:** Gọi hàm `fibonacci(3)`

Xét điều kiện  $3 \leq 1$  sai , thực hiện điều kiện của else : `return fibonacci(2) + fibonacci(1)`

**Bước 7:** Gọi hàm `fibonacci(2)`

Xét điều kiện  $2 \leq 1$  sai , thực hiện điều kiện của else : `return fibonacci(1) + fibonacci(0)`

**Bước 8:** Gọi hàm `fibonacci(1)`

Xét điều kiện  $1 \leq 1$  đúng , thực hiện điều kiện của if : trả về 1

**Bước 9:** Gọi hàm `fibonacci(0)`

Xét điều kiện  $0 \leq 1$  đúng , thực hiện điều kiện của if : trả về 0

**Thực hiện lần lượt từ bước 9 lên bước 1 :**

$\text{fibonacci}(0) = 0 \rightarrow \text{fibonacci}(1) = 1 \rightarrow \text{fibonacci}(2) = 0 + 1 = 1 \rightarrow \text{fibonacci}(3) = 1 + 1 = 2 \rightarrow \text{fibonacci}(4) = 2 + 1 = 3 \rightarrow \text{fibonacci}(5) = 2 + 3 = 5 \rightarrow \text{fibonacci}(6) = 5 + 3 = 8 \rightarrow \text{fibonacci}(7) = 8 + 5 = 13 \rightarrow \text{fibonacci}(8) = 13 + 8 = 21 \rightarrow$  kết thúc chương trình

**Vậy số fibonacci thứ 8 là 21**

### Bài 3 :

**Bước 1 :** Gọi hàm `power(2, 6)`

Xét điều kiện `6 == 0` sai , thực hiện câu lệnh của `else` : `return 2 * pow(2, 5)`

**Bước 2 :** Gọi hàm `power(2, 5)`

Xét điều kiện `5 == 0` sai , thực hiện câu lệnh của `else` : `return 2 * pow(2, 4)`

**Bước 3 :** Gọi hàm `power(2, 4)`

Xét điều kiện `4 == 0` sai , thực hiện câu lệnh của `else` : `return 2 * pow(2, 3)`

**Bước 4 :** Gọi hàm `power(2, 3)`

Xét điều kiện `3 == 0` sai , thực hiện câu lệnh của `else` : `return 2 * pow(2, 2)`

**Bước 5 :** Gọi hàm `power(2, 2)`

Xét điều kiện `2 == 0` sai , thực hiện câu lệnh của `else` : `return 2 * pow(2, 1)`

**Bước 6 :** Gọi hàm `power(2, 1)`

Xét điều kiện `1 == 0` sai , thực hiện câu lệnh của `else` : `return 2 * pow(2, 0)`

**Bước 7 :** Gọi hàm `power(2, 0)`

Xét điều kiện `0 == 0` đúng, trả về kết quả là 1

**Thực hiện từ bước 7 lên bước 1 ta được :**

$\text{power}(2, 0) = 1 \rightarrow \text{power}(2, 1) = 2 * 1 = 2 \rightarrow \text{power}(2, 2) = 2 * 2 = 4 \rightarrow \text{power}(2, 3) = 2 * 4 = 8 \rightarrow \text{power}(2, 4) = 2 * 8 = 16 \rightarrow \text{power}(2, 5) = 2 * 16 = 32 \rightarrow \text{power}(2, 6) = 2 * 32 = 64 \rightarrow$  kết thúc chương trình

**Vậy kết quả của 2 mũ 6 là 64**

## Bài 4 :

**Bước 1 :** Gọi hàm `thap_ha_noi(4, A, B, C)`

Xét điều kiện  $4 == 1$  sai, thực hiện câu lệnh của else: `thap_ha_noi(3, A, B, C)`, chuyển 3 đĩa trên cùng của cột A sang cột C lấy cột B làm trung gian

In ra: Chuyển đĩa 4 từ cột A sang cột B

- **1.1 :** `thap_ha_noi(3, C, A, B)`, chuyển 3 đĩa từ cột C sang B và dùng A làm trung gian

**Bước 2 :** Gọi hàm `thap_ha_noi(3, A, B, C)`

Xét điều kiện  $3 == 1$  sai, thực hiện câu lệnh của else: `thap_ha_noi(2, A, B, C)`, chuyển 2 đĩa trên cùng của cột A sang cột B lấy cột C làm trung gian

In ra: Chuyển đĩa 3 từ cột A sang cột C

`thap_ha_noi(2, B, C, A)`, chuyển 2 đĩa từ cột B sang C sử dụng cột A làm trung gian

- **2.1 :** `thap_ha_noi(3, C, A, B)`, chuyển 2 đĩa trên cùng từ cột C sang cột A sử dụng cột B làm trung gian

In ra : Chuyển đĩa thứ 3 từ cột C sang cột B

thap\_ha\_noi(2, A, B, C), chuyển 2 đĩa từ cột A sang B sử dụng cột C làm trung gian

**Bước 3 :** Gọi hàm `thap_ha_noi(2, A, B, C)`

Xét điều kiện  $2 == 1$  sai, thực hiện câu lệnh của `else`: `thap_ha_noi(1, A, B, C)`, chuyển đĩa trên cùng của cột A sang cột C lấy cột B làm trung gian

In ra : Di chuyển đĩa thứ 2 từ cột A sang cột B  
`thap_ha_noi(1, C, B, A)`, di chuyển đĩa trên cùng từ cột C sang cột B sử dụng cột A làm trung gian

- **3.1 :** `thap_ha_noi(1, B, A, C)`, di chuyển từ cột B sang A sử dụng cột C làm trung gian

In ra : Chuyển đĩa 2 từ cột B sang C  
`thap_ha_noi(1, A, C, B)`, chuyển đĩa trên cùng từ cột A sang C sử dụng B làm cột trung gian

- **3.2 :** `thap_ha_noi(1, C, B, A)`, chuyển đĩa trên cùng từ cột C sang B sử dụng cột A làm trung gian

In ra : Chuyển đĩa 2 từ cột C sang A  
`thap_ha_noi(1, B, A, C)`, chuyển đĩa trên cùng từ cột B sang cột A sử dụng C làm trung gian

- **3.3 :** `thap_ha_noi(1, A, B, C)`, chuyển đĩa trên cùng từ cột A sang cột B sử dụng cột C làm trung gian

In ra : Chuyển đĩa 2 từ cột A sang cột B  
`thap_ha_noi(1, C, B, A)`, chuyển đĩa trên cùng từ cột C sang cột B sử dụng cột A làm trung gian

**Bước 4 :** Gọi hàm `thap_ha_noi(1, A, B, C)`

Xét điều kiện  $1 == 1$  đúng nên in ra : Chuyển đĩa 1 từ cột A sang cột B

## Bài 5 :

**Bước 1 :** Gọi hàm cho\_ga(36, 100)

Xét điều kiện  $36 == 0$  và  $100 == 0$  là sai, chuyển xuống thực hiện vòng lặp if :  $100 \% 2 != 0$  là sai, chuyển xuống thực hiện vòng lặp tiếp theo

**Bước 2 :** Vòng lặp kiểm tra số lượng chó: for cho in range(37), kiểm tra từ 0 đến 36 con

Khởi tạo biến ga = 100 - cho

Nếu  $ga * 2 + cho * 4 == 100$  thì trả về kết quả của ga, cho

Nếu sai , thoát vòng lặp

**Bước 3 :** Gọi hàm đệ quy cho\_ga(35, 96)

Nếu  $ga != -1$  đúng thì trả về cho + 1 đơn vị , nếu sai thì trả về -1, -1

Giả sử chó = 7, gà = 29  $\Rightarrow 29 * 2 + 7 * 4 = 86 != 100$  nên loại

Đệ quy : Giảm gà xuống 28 , chó tăng lên 1  $\Rightarrow 28 * 2 + 8 * 4 = 88 != 100 \Rightarrow$  loại

Thực hiện lần lượt đến gà = 22, chó = 14 :  $22 * 2 + 14 * 4 = 100 \Rightarrow$  đúng  $\Rightarrow$  in ra kết quả là chó: 14, gà : 22

**Vậy số chó tìm được là 14 và gà là 22**