**HÀ HUY HIẾU\_DHKL**17A2\_2174600119-Ca sáng

Câu 1

Bước 1: Gọi hàm sum\_of\_numbers(7) Kiểm tra điều kiện cơ sở: n=7, không thỏa mãn điều kiện cơ sở

Thực hiện câu lệnh trong else: return 7 + sum\_of\_numbers(6)

Bước 2: Hàm sum\_of\_numbers(6) được gọi Kiểm tra điều kiện cơ sở: n=6, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 6 + sum\_of\_numbers(5)

Bước 3: Hàm sum\_of\_numbers(5) được gọi Kiểm tra điều kiện cơ sở: n=5, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 5 + sum\_of\_numbers(4)

Bước 4: Hàm sum\_of\_numbers(4) được gọi Kiểm tra điều kiện cơ sở: n=4, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 4 + sum\_of\_numb

Bước 5: Hàm sum\_of\_numbers(3) được gọi Kiểm tra điều kiện cơ sở: n=3, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 3 + sum\_of\_numbers(2)

Bước 6: Hàm sum\_of\_numbers(2) được gọi Kiểm tra điều kiện cơ sở: n=2, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 2 + sum\_of\_numbers(1)

Bước 7: Hàm sum\_of\_numbers(1) được gọi Kiểm tra điều kiện cơ sở: n=1, điều kiện cơ sở được thỏa mãn

Ta được các giá trị là :

sum\_of\_numbers(1) -> 1

sum\_of\_numbers(2) -> 2 + 1 = 3

sum\_of\_numbers(3) -> 3 + 3 = 6

sum\_of\_numbers(4) ->4 + 6 = 10

sum\_of\_numbers(5)-> 5 + 10 = 15

sum\_of\_numbers(6)-> 6 + 15 = 21

sum\_of\_numbers(7)-> 7 + 21 = 28

kết quả là 28

Câu 2:

**HÀ HUY HIẾU\_DHKL**17A2\_2174600119-Ca sáng

1. được gọi **fibonacci(8)**

**n** không nhỏ hơn 1, do đó chúng ta sẽ đi vào phần **else**.

hàm sẽ trả về **fibonacci(7) + fibonacci(6)**.

1. được gọi **fibonacci(7)**

**điều kiện thỏa mãn**

**n** không nhỏ hơn 1, do đó chúng ta sẽ đi vào phần **else**.

Hàm sẽ trả về **fibonacci(6) + fibonacci(5)**.

1. được gọi **fibonacci(6)**

**điều kiện**

**n** không nhỏ hơn 1, do đó chúng ta sẽ đi vào phần **else**.

Hàm sẽ trả về **fibonacci(5) + fibonacci(4)**.

1. được gọi **fibonacci(5)**

**điều kiện thỏa mãn**

**n** không nhỏ hơn 1, do đó chúng ta sẽ đi vào phần **else**.

Hàm sẽ trả về **fibonacci(4) + fibonacci(3)**.

1. được gọi **fibonacci(4)**

**điều kiện**

**n** không nhỏ hơn 1, do đó chúng ta sẽ đi vào phần **else**.

Hàm sẽ trả về **fibonacci(3) + fibonacci(2)**.

1. được gọi **fibonacci(3)**

**n** không nhỏ hơn 1, do đó chúng ta sẽ đi vào phần **else**.

Hàm sẽ trả về **fibonacci(2) + fibonacci(1)**.

1. được gọi **fibonacci(2)**

**n** không nhỏ hơn 1, do đó chúng ta sẽ đi vào phần **else**.

Hàm sẽ trả về **fibonacci(1) + fibonacci(0)**.

1. được gọi **fibonacci(1)**

Khi **n = 1**, hàm trả về giá trị 1 (trường hợp cơ sở).

1. được gọi: **fibonacci(0)**

Khi **n = 0**, hàm trả về giá trị 0 (trường hợp cơ sở).

Các hàm ta được là :

**fibonacci(1)** trả về 1.

**fibonacci(2)** trả về **fibonacci(1) + fibonacci(0)** = 1.

**fibonacci(3)** trả về **fibonacci(2) + fibonacci(1)** = 2.

**fibonacci(4)** trả về **fibonacci(3) + fibonacci(2)** = 3.

**fibonacci(5)** trả về **fibonacci(4) + fibonacci(3)** = 5.

**fibonacci(6)** trả về **fibonacci(5) + fibonacci(4)** = 8.

**fibonacci(7)** trả về **fibonacci(6) + fibonacci(5)** = 13.

**fibonacci(8)** trả về **fibonacci(7) + fibonacci(6)** = 21.

Tất cả số fibonacci(8) là giá trị 21

Câu 3

**HÀ HUY HIẾU\_DHKL**17A2\_2174600119-Ca sáng

Bước 1: Gọi hàm power(2, 6) Kiểm tra điều kiện cơ sở: n=6, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 2 \* power(2, 5)

Bước 2: Hàm power(2, 5) được gọi Kiểm tra điều kiện cơ sở: n=5, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 2 \* power(2, 4)

Bước 3: Hàm power(2, 4) được gọi Kiểm tra điều kiện cơ sở: n=4, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 2 \* power(2, 3)

Bước 4: Hàm power(2, 3) được gọi Kiểm tra điều kiện cơ sở: n=3, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 2 \* power(2, 2)

Bước 5: Hàm power(2, 2) được gọi Kiểm tra điều kiện cơ sở: n=2 không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 2 \* power(2, 1)

Bước 6: Hàm power(2, 1) được gọi Kiểm tra điều kiện cơ sở: n=1, không thỏa mãn điều kiện cơ sở Thực hiện câu lệnh trong else: return 2 \* power(2, 0)

Bước 7: Hàm power(2, 0) được gọi Kiểm tra điều kiện cơ sở: n=0, thỏa mãn điều kiện cơ sở. Hàm trả về 1.

**power(2, 1)** -> **2** power(2, 0) = 2 \* 1 = 2.

**power(2, 2)** ->2 power(2, 1) = 2 \* 2 = 4.

**power(2, 3)** -> 2 power(2, 2) = 2 \* 4 = 8.

**power(2, 4)** -> 2 power(2, 3) = 2 \* 8 = 16.

**power(2, 5)** -> 2 power(2, 4) = 2 \* 16 = 32.

**power(2, 6)** ->2 power(2, 5) = 2 \* 32 = 64.

Giá trị 26=64.

Câu 4

**HÀ HUY HIẾU\_DHKL**17A2\_2174600119-Ca sáng

1: Gọi hàm thap\_ha\_noi(4, “A”, “C”, “B”) Bắt đầu với 4 đĩa ở cột A, chuyển chúng sang cột B, sử dụng cột C làm cột trung gian.

2: Gọi hàm thap\_ha\_noi(3, “A”, “B”, “C”) Chuyển 3 đĩa từ cột A sang cột C, sử dụng cột B làm cột trung gian

3: Gọi hàm thap\_ha\_noi(2, “A”, “C”, “B”) Chuyển 2 đĩa từ cột A sang cột B, sử dụng cột C làm trung gian  
 4: Gọi hàm thap\_ha\_noi(1, “A”, “B”, “C”) Có 1 đĩa ở cột A, di chuyển nó sang cột B Hàm trả về dòng in “Chuyển đĩa 1 từ cột A sang cột B”

5:Sau khi thực hiện bước 4, chúng ta quay lại bước 3, nơi chúng ta thực hiện in ra câu lệnh "Di chuyển đĩa 2 từ cột A sang cột B".

6:Tiếp tục quay lại các bước trước đó, chúng ta in ra các câu lệnh tương tự để chuyển các đĩa còn lại từ cọc A sang cọc B.

7:Quá trình này tiếp tục cho đến khi chúng ta đã chuyển tất cả các đĩa từ cọc A sang cọc B.

**Ta được :chuỗi các câu lệnh mô tả các bước cần thiết để chuyển 4 đĩa từ cọc A sang cọc B theo quy tắc** của bài toán Tháp Hà Nội.

Bài 5

**HÀ HUY HIẾU\_DHKL**17A2\_2174600119-Ca sáng

Gọi **count\_animals(3, 10)**

1.Nếu cả tong\_so\_con và tong\_so\_chan đều bằng 0, có nghĩa là không có con thú nào, hàm trả về (0, 0). Nếu tong\_so\_chan không chia hết cho 2 thì ta số chân không thể tạo thành số lượng con thú nguyên tắc->-1, -1

2. dựa trên số lần lặp của số lượng gà và chó-> ta tính số chân gà và chó

3.vòng lặp for hàm gọi đệ quy với số con thú giảm đi 1 và số chân giảm đi 4 -> tổng số chân gà và chó là 4

Vì vậy sẽ là thông báo Số chân không hợp lệ để tạo ra con vật. vì số chân không chia hết cho 2, không thể tạo ra con vật từ số chân đã cho.

Top of Form

Top of Form

Top of Form

Top of Form