Mục lục

Chương 9

Hệ thức đệ quy

Hệ thức đệ quy cấp k của dãy (a_n) có dạng

$$F(a_n, a_{n-1}, a_{n-2}, \dots, a_{n-k}, n) = 0, \ \forall n \geq k, \tag{9.1}$$

mô tả mối liên hệ giữa một phần tử của dãy với k phần tử đứng trước nó. Thông thường, a_n biểu diễn duy nhất theo $a_{n-1}, a_{n-2}, \dots, a_{n-k}$ và n, và khi đó, dãy hoàn toàn xác định khi biết k phần tử ban đầu, thường là $a_0, a_1, \dots, a_{k-1}$.

Ta thường gặp hệ thức đệ quy có dạng

$$c_0 a_n + c_1 a_{n-1} + c_2 a_{n-1} + \cdots + c_k a_{n-k} = f(n), \tag{9.2}$$

trong đó $c_0 \neq 0$, tức là a_n biểu diễn tuyến tính theo các phần tử đứng trước nó. Ta gọi đó là hệ thức đệ quy tuyến tính. Nếu $f(n) = 0$, hệ thức đệ quy tuyến tính này gọi là thuần nhất. Ở đây, các hệ số $c_0, c_1, \dots, c_k$ có thể là hằng số hoặc phụ thuộc n.

Trong Python, để giải hệ thức đệ quy, ta thực hiện các bước sau

1) Khai báo thư viện SymPy, biến n, và dãy số a_n như một hàm

```
from sympy import *
n = symbols('n')
a = symbols('a', cls=Function)
```

2) Các lệnh giải hệ thức đệ quy, trước hết là giải chưa có điều kiện ban đầu, thì nghiệm sẽ chứa các hằng số bất định $C_0, C_1, \dots, C_{k-1}$.

```
rsolve( F , a(n) )
```

Giải hệ thức với điều kiện ban đầu $a_{i_1} = v_1, a_{i_2} = v_2, \dots, a_{i_k} = v_k$, ta dùng

```
rsolve( F , a(n) , {a(i1) : v1 , a(i2) : v2 ,... , a(ik) : vk} )
```

hoặc

```
rsolve( F , a(n) , {i1 : v1 , i2 : v2 ,... , ik : vk} )
```

Trong trình bày sau này, để ngắn gọn, các lệnh khai báo như dòng 1–3 sẽ được lược bớt. Vì vậy, muốn mã hoàn chỉnh và chạy được, ta cần bổ sung lại các khai báo này.

9.1 Hệ thức đệ quy tuyến tính cấp một

Hệ thức đệ quy tuyến tính cấp một tổng quát

$$a_n = c_n a_{n-1} + f_n \tag{9.3}$$

có nghiệm

$$a_n = c_1 c_2 \cdots c_n \Big(a_0 + \frac{f_1}{c_1} + \frac{f_2}{c_1 c_2} + \frac{f_3}{c_1 c_2 c_3} + \cdots + \frac{f_n}{c_1 c_2 \cdots c_n} \Big). \tag{9.4}$$

Trường hợp đặc biệt

Cấp số cộng: $a_n = a_{n-1} + d$ ứng với $c_n = 1$, $f_n = d$, có nghiệm $a_n = a_0 + nd$.

Cấp số nhân: $a_n = q a_{n-1}$ ứng với $c_n = q$, $f_n = 0$, có nghiệm $a_n = a_0 q^n$.

Hầu hết hệ thức (9.3) với c_n không phải hằng số đều không giải được bằng Python. Tuy nhiên, có thể giải cấp số nhân bằng lệnh

```
rsolve( a(n+1) - q*a(n) , a(n) , {a(0): a0} )   # → a0 q^n
```

Hai ví dụ về toán tài chính.

Ví dụ 9.1. Một ngân hàng trả lãi kép 6% hàng năm cho tài khoản tiết kiệm. Nếu ban đầu, tài khoản có 100 triệu, thì sau 10 năm là bao nhiêu?

Giải. Đặt a_n là số tiền (đơn vị: triệu) trong tài khoản sau n năm. Khi đó, $a_0 = 100$, và

$$a_{n+1} = a_n + 6\% \times a_n = 1.06 a_n, \quad \forall n \geq 0.$$

Do đó $a_n = 100 \cdot 1.06^n$, và suy ra $a_{10} = 179.085$. □

Ví dụ 9.2. An vay một khoản tiền S với lãi suất tháng r, và phải trả trong T tháng –không tính tháng vay. Tìm số tiền cố định P mà An phải trả hàng tháng.

Giải. Đặt a_n là số tiền còn nợ vào cuối tháng của lần trả thứ n. Khi đó, vào cuối tháng thứ $n+1$, số tiền An còn nợ là a_n (số tiền nợ trong tháng n) $+ra_n$ (lãi suất) $-P$ (số tiền trả vào cuối tháng thứ $(n+1)$. Ta có hệ thức đệ quy

$$a_{n+1} = a_n + ra_n - P = (1+r)\,a_n - P,$$

trong đó $a_0 = S$. Suy ra $a_n = \frac{P}{r} + (r+1)^n\Big(S - \frac{P}{r}\Big)$.

An muốn $a_T = 0$, nên $P = \frac{Sr(r+1)^T}{(r+1)^T - 1} = \frac{Sr}{1-(1+r)^{-T}}$. □

Ví dụ 9.3. Đặt a_n là số các tổng riêng của $n \in \mathbb{Z}^+$. Lập hệ thức đệ quy và giải a_n.

Giải. Để tìm quan hệ đệ quy, ta chia a_{n+1} tổng riêng của $n+1$ thành hai phần:

1) Số hạng cuối lớn hơn 1. Khi đó, nếu trừ số hạng cuối đi 1, ta lại được một tổng riêng của n, tức là, mỗi tổng riêng dạng này tương ứng 1–1 với một tổng riêng của n. Suy ra phần này có a_n tổng riêng.

2) Số hạng cuối là 1. Khi đó, các số hạng đầu là tổng riêng của n, gồm a_n tổng riêng.

Do đó $a_{n+1} = a_n + a_n = 2a_n$, $\forall n \geq 1$, với $a_1 = 1$. Vậy $a_n = 2^{n-1}$, $\forall n \geq 1$. □

Bảng sau mô tả chi tiết lập luận trên, trong trường hợp xây dựng các tổng riêng của 4 từ các tổng riêng của 3.

		(1’)	4
		(2’)	1 + 3
(1)	3	(3’)	2 + 2
(2)	1 + 2	(4’)	1 + 1 + 2
(3)	2 + 1		
(4)	1 + 1 + 1	(1”)	3 + 1
		(2”)	1 + 2 + 1
		(3”)	2 + 1 + 1
		(4”)	1 + 1 + 1 + 1

```python
def summands(n):
    if n == 1:
        return [[1]]

    A = []
    for a in summands(n-1):
        b = a.copy()
        a[-1] += 1
        A.append(a)

        b.append(1)
        A.append(b)
    return A

summands(3)  # [[3], [2, 1], [1, 2], [1, 1, 1]]
```

Ví dụ 9.4. Thuật toán *sắp xếp nổi bọt*:

```python
def BubbleSort(x):                    # x = [x_0, x_1, ..., x_{n-1}]
  n = len(x)                          # độ dài của x
  for i in range(n-1):                # duyệt từ đầu, x_0, tới gần cuối, x_{n-2}.
    for j in range(n-1,i,-1):# duyệt từ cuối, x_{n-1} về kế sau x_i, tức x_{i+1}
      if x[j] < x[j-1]:
        x[j-1], x[j] = x[j], x[j-1]   # đổi chỗ
  return x                            # trả về kết quả cho hàm

BubbleSort([7, 9, 2, 5, 8])           # kết quả [2, 5, 7, 8, 9]
```

Đặt a_n là số phép so sánh –cũng là số chu trình tối giản– của thuật toán khi sắp xếp dãy n phần tử. Lập hệ thức đệ quy và giải a_n.

Giải. Thuật toán gồm hai giai đoạn:

1) Ứng với $i = 0$, kiểm tra $n-1$ phép so sánh $x_j < x_{j-1}$, với $j = \overline{n-1\downarrow 1}$, và thực hiện phép đổi chỗ nếu cần. Sau bước này, $x_0 \le x_i,\ \forall i > 0$.

2) Thực hiện thuật toán nổi bọt cho dãy $n-1$ phần tử $x_1, x_2, \ldots, x_{n-1}$, mà số phép so sánh là a_{n-1}, theo định nghĩa.

Như vậy, $a_n = (n-1) + a_{n-1}$, và $a_1 = 0$. Do đó $a_n = \dfrac{n(n-1)}{2}$. □

```python
rsolve( -a(n) + (n-1)+a(n-1) , a(n) , {a(1): 0}).simplify()
```

Ta mô tả chi tiết thuật toán với dãy $x = (7, 9, 2, 5, 8)$ bởi hình sau

$i = 0$	x_0	7		7		7		7	$j = 1$	2
	x_1	9		9		9	$j = 2$	2		7
	x_2	2		2	$j = 3$	2		9		9
	x_3	5	$j = 4$	5		5		5		5
	x_4	8		8		8		8		8
	Bốn phép so sánh và hai phép đổi chỗ									
$i = 1$	x_0	2		2		2		2		
	x_1	7		7		7	$j = 2$	5		
	x_2	9		9	$j = 3$	5		7		
	x_3	5	$j = 4$	5		9		9		
	x_4	8		8		8		8		
	Ba phép so sánh và hai phép đổi chỗ									
$i = 2$	x_0	2		2		2				
	x_1	5		5		5				
	x_2	7		7	$j = 3$	7				
	x_3	9	$j = 4$	8		8				
	x_4	8		9		9				
	Hai phép so sánh và một phép đổi chỗ									
$i = 3$	x_0	2								
	x_1	5								
	x_2	7								
	x_3	8	$j = 4$							
	x_4	9								
	Một phép so sánh và không có phép đổi chỗ									

Ví dụ 9.5. Đặt a_n là số hoán vị của n vật, đánh số từ 1 tới n. Lập hệ thức đệ quy và giải a_n.

Giải. Từ mỗi hoán vị của $n-1$ vật $1, 2, \dots, n-1$, ta tạo ra hoán vị của n vật bằng cách xếp vật thứ n vào trước, sau, hoặc chèn vào giữa hoán vị của $n-1$ vật này. Như vậy, có n vị trí để xếp vật thứ n. Mặt khác, theo định nghĩa, số hoán vị của $n-1$ vật là a_{n-1}, nên theo quy tắc nhân $a_n = na_{n-1}$, $\forall n \geq 2$. Với $a_1 = 1$, ta tìm được $a_n = n!$. □

Chẳng hạn, cách sinh hoán vị của $\{1, 2\}$ từ $\{1\}$:

$$\begin{array}{ccc} 2 & 1 & \\ & 1 & 2 \end{array}$$

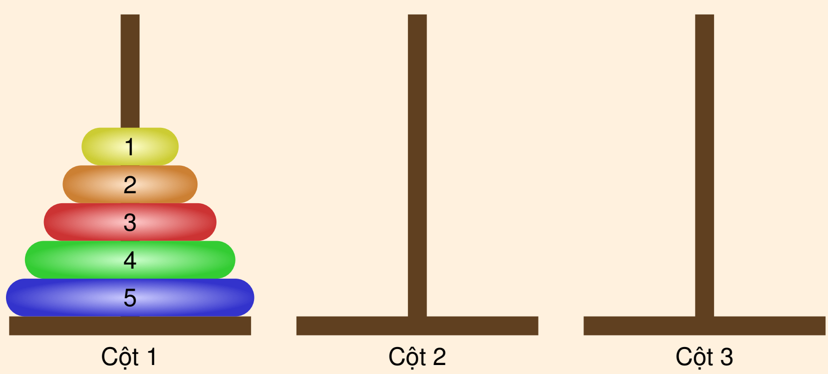
và các hoán vị của $\{1, 2, 3\}$ từ $\{1, 2\}$:

3	1		2	
	1	3	2	
	1		2	3
3	2		1	
	2	3	1	
	2		1	3

```python
def permutations(n):
  if n == 1:
    return [[1]]
  A = []
  for a in permutations(n-1):
    for i in range(n):
        b = a.copy()
        b.insert(i, n)
        A.append(b)
  return A

permutations(3)  # → [[3, 2, 1], [2, 3, 1], [2, 1, 3], [3, 1, 2], [1, 3, 2], [1, 2, 3]]
```

Ví dụ 9.6 (Tháp Hà Nội)**.** Xét n đĩa tròn (có đường kính khác nhau) có lỗ ở tâm được xếp chồng lên nhau trên như hình dưới. Trong hình, $n = 5$ và các đĩa xếp ở cột 1 mà đĩa trên nhỏ hơn đĩa dưới. Việc di chuyển các đĩa từ cột này sang cột kia phải thỏa mãn: (1) mỗi lần chỉ chuyển một đĩa, và (2) tại mỗi cột, đĩa trên nhỏ hơn đĩa dưới. Dùng các cột 1, 2, và 3 làm vị trí tạm thời cho các đĩa, ta cần chuyển các hết đĩa sang cột 3.



Đặt a_n là số lần chuyển *ít nhất* để chuyển n đĩa từ cột 1 sang cột 3. Lập hệ thức đệ quy và giải a_n.

Giải. Khi đó, với $n + 1$ đĩa, ta có thể làm như sau:

a) Chuyển n đĩa trên cùng từ cột 1 sang cột 2. Việc này cần ít nhất a_n lần chuyển.

b) Chuyển đĩa to nhất từ cột 1 sang cột 3. Có một lần chuyển.

c) Cuối cùng, chuyển n đĩa từ cột 2 lên trên đĩa to nhất ở cột 3. Việc này cần ít nhất a_n lần chuyển.

Do đó, $a_{n+1} \leq a_n + 1 + a_n = 2a_n + 1$.

Mặt khác, với $n + 1$ đĩa, tại thời điểm nào đó, đĩa to nhất (đĩa dưới cùng tại cột 1) phải được chuyển sang cột 3. Lúc này, cột 3 không có đĩa nào, và n đĩa nhỏ hơn đã chuyển sang cột 2. Để chuyển n đĩa này, cần ít nhất a_n lần chuyển. Đĩa to nhất cần di chuyển ít nhất một lần để sang được cột 3. Sau đó, để đặt n đĩa nhỏ lên trên đĩa to nhất (đều trên cột 3), cần ít nhất a_n bước nữa. Vậy, $a_{n+1} \geq a_n + 1 + a_n = 2a_n + 1$.

Suy ra, $a_{n+1} = 2a_n + 1$. Với $a_1 = 1$, ta có $a_n = 2^n - 1$. □

Giả sử một máy tính mô phỏng các lần chuyển, được một tỷ lần chuyển mỗi giây. Khi đó để mô phỏng các lần chuyển cho hệ 64 đĩa, máy tính cần

$$\frac{2^{64} - 1}{10^9} \text{ giây } = 18\,446\,744\,074 \text{ giây } = \frac{18\,446\,744\,074}{3600 \times 24 \times 365} \approx 585 \text{ năm!}$$

Ví dụ 9.7. Ký hiệu a_n là số xâu tứ phân độ dài n có một số chẵn các số 1. Bằng cách xét ký tự đầu, lập hệ thức đệ quy và giải a_n.

Giải. Trong các xâu đếm bởi a_n, xét hai trường hợp:

1) Ký tự đầu là 1, thì xâu gồm $n - 1$ ký tự còn lại phải có một số lẻ các số 1. Số xâu như vậy bằng số xâu độ dài $n - 1$, trừ đi số xâu, vẫn độ dài $n - 1$, có một số chẵn các số 1, tức là $4^{n-1} - a_{n-1}$.

2) Ký tự đầu là 0, 2, hoặc 3. Khi đó xâu $n - 1$ ký tự còn lại có một số chẵn các số 1, gồm a_{n-1} xâu. Theo quy tắc nhân, trường hợp này có $3a_{n-1}$ xâu.

Theo quy tắc cộng, $a_n = (4^{n-1} - a_{n-1}) + 3a_{n-1} = 2a_{n-1} + 4^{n-1}$, $n \geq 2$. Với $a_1 = 3$, ta có $a_n = \dfrac{2^n}{2} + \dfrac{4^n}{2} = 2^{n-1}(2^n + 1)$. □

Vì mỗi xâu là một hoán vị lặp của bốn số 0, 1, 2, và 3, nên có thể dùng hàm sinh mũ $f(x) = \sum_{n=0}^{\infty} a_n \dfrac{x^n}{n!}$ để giải bài toán này. Ta có

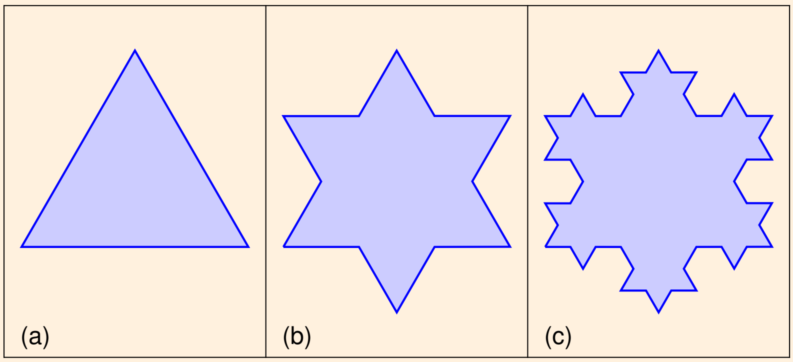
$$f(x) = \underbrace{\left(1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \ldots\right)}_{\text{chọn 1}} \underbrace{\left(1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots\right)^3}_{\text{chọn 0, 2, hoặc 3}}$$

$$= e^{3x}\cosh x = e^{3x}\frac{e^x + e^{-x}}{2} = \frac{1}{2}(e^{4x} + e^{2x}) = \frac{1}{2}\left[\sum_{n=0}^{\infty}\frac{(4x)^n}{n!} + \sum_{n=0}^{\infty}\frac{(2x)^n}{n!}\right].$$

Vậy $a_n = \frac{1}{2}(4^n + 2^n)$.

Tiếp theo, một ví dụ về hình học Fractal.

Ví dụ 9.8 ("Bông tuyết" Koch)**.** *Cho với tam giác đều cạnh bằng 1, như phần (a) của hình dưới, được biến đổi thành sao David trong phần (b) bằng cách chia mỗi cạnh thành ba đoạn bằng nhau, bỏ đi đoạn ở giữa và gắn một tam giác đều mới, về phía ngoài, tại cạnh vừa bỏ đi. Tiếp tục quá trình này, ta biến đổi sao David thành đa giác ở phần (c).



Với $n \geq 0$, đặt a_n là diện tích của đa giác P_n thu được từ tam giác đều ban đầu sau n phép biến đổi được mô tả ở trên.

Lập hệ thức đệ quy và giải a_n.

Giải. Ta có

$$\begin{aligned} a_{n+1} &= a_n + \text{ tổng diện tích các tam giác gắn thêm} \\ &= a_n + \text{ số tam giác gắn thêm } \times \text{ diện tích mỗi tam giác gắn thêm} \end{aligned}$$

Đặt e_n, l_n là số cạnh, độ dài một cạnh của P_n. Khi đó

$$\begin{aligned} e_{n+1} &= 4e_n, \text{ với } e_0 = 3, \text{ và} \\ l_{n+1} &= \frac{l_n}{3}, \text{ với } l_0 = 1. \end{aligned}$$

Suy ra $e_n = 3 \cdot 4^n$, và $l_n = \frac{1}{3^n}$. Do đó

$$a_{n+1} = a_n + e_n \times \frac{\sqrt{3}}{4}l_{n+1}^2 = a_n + 3 \cdot 4^n\frac{\sqrt{3}}{4}\left(\frac{1}{3^{n+1}}\right)^2 = a_n + \frac{1}{4\sqrt{3}}\left(\frac{4}{9}\right)^n.$$

*Bông tuyết Koch được đưa ra năm 1904, bởi nhà toán học Thụy Điển Helge von Koch (1870–1924), với diện tích hữu hạn nhưng chu vi vô hạn.

Giải hệ thức đệ quy với $a_0 = \dfrac{\sqrt{3}}{4}$, ta được $a_n = \dfrac{2\sqrt{3}}{5} - \dfrac{3\sqrt{3}}{20}\left(\dfrac{4}{9}\right)^n$. □

Phương pháp đệ quy là lĩnh vực cơ bản của toán rời rạc và phân tích thuật toán. Phương pháp này nảy sinh khi ta muốn giải bài toán bằng cách chia nhỏ, hoặc đưa nó về các bài toán tương tự nhưng cỡ nhỏ hơn. Trong nhiều ngôn ngữ lập trình, điều này được thực hiện bằng cách sử dụng hàm và thủ tục đệ quy, theo quy cách được phép gọi chính nó.

Ví dụ 9.9. Xây dựng và viết mã thuật toán đệ quy tính

a) a^n, với $a \in \mathbb{R}$, $n \in \mathbb{N}$.

b) Ước chung lớn nhất $\gcd(a, b)$ của hai số tự nhiên không đồng thời bằng 0.

Giải. a) Thuật toán đệ quy tính a^n, với $a \in \mathbb{R}$, $n \in \mathbb{N}$.

1) Với $a^0 = 1$, và $a^n = a \cdot a^{n-1}$, $\forall n \geq 1$, ta có thuật toán đệ quy

Mã 1:

```
def luy_thua(a, n):
  if n == 0:
    return 1
  return a * luy_thua(a, n-1)

luy_thua(2, 10)  # → 1024
```

Mã 2:

```
luy_thua = lambda a, n: 1 if n==0 else a * luy_thua(
    a, n-1)
```

2) Mặt khác, nếu xét phép chia n cho 2, $n = 2n' + r$, trong đó $r = 0$ hoặc 1. Khi đó

$$a^n = a^{2n'+r} = a^r \cdot (a^2)^{n'},$$

ta được phương pháp chia đôi

Mã 1:

```
def luy_thua(a, n):
  if n == 0:
    return 1
  return a**(n%2) * luy_thua(a*a, n//2) # n%2 là r,
    n//2 là n'
```

Mã 2:

```
luy_thua = lambda a, n: 1 if n==0 else a**(n%2) *
    luy_thua(a*a, n//2)
```

b) Nếu $b = 0$, thì $a \neq 0$, và $\gcd(a, b) = a$. Ngược lại, xét phép chia a cho b, $a = bq + r$, trong đó $q, r \in \mathbb{N}$, $r < b$, thì

$$\gcd(a, b) = \gcd(b, r) = \gcd(b, a \bmod b).$$

Ta được thuật toán đệ quy

Mã 1:

```
def gcd(a, b):
  if b == 0:
    return a
  else:
    return gcd(b, a % b)

gcd(333, 84)  # → 3
```

Mã 2:

```
gcd = lambda a, b: a if b==0 else gcd(b, a % b)
```

□

Bài tập 9.1

9.1. Tìm quan hệ đệ quy của các cấp số nhân sau

a) $2, 10, 50, 250, \ldots$

b) $6, -18, 54, -162, \ldots$

c) $7, \frac{14}{5}, \frac{28}{25}, \frac{56}{125}, \ldots$

9.2. Giải các hệ thức đệ quy

a) $a_{n+1} - 1.5a_n = 0$

b) $4a_n - 5a_{n-1} = 0$

c) $3a_{n+1} - 4a_n = 0$, $a_1 = 5$

d) $2a_n - 3a_{n-1} = 0$, $a_4 = 81$

9.3. Cho quan hệ đệ quy $a_{n+1} - da_n = 0$. Biết $a_3 = \frac{153}{49}$ và $a_5 = \frac{1377}{240}$, tìm d.

9.4.

9.5.

9.2 Hệ thức đệ quy tuyến tính thuần nhất cấp cao hệ số hằng

Ví dụ 9.10. Giải các hệ thức đệ quy

a) $a_n + a_{n-1} - 6a_{n-2} = 0,\ \forall n \geq 2$, trong đó $a_0 = -1, a_1 = 8$.

b) $a_{n+2} = 4a_{n+1} - 4a_n,\ \forall n \geq 0$, trong đó $a_0 = 1, a_1 = 3$.

c) $2a_{n+3} = a_{n+2} + 2a_{n+1} - a_n,\ \forall n \geq 0$, trong đó $a_0 = 0, a_1 = 1, a_2 = 2$.

Giải. a) $a_n = 2^n - 2(-3)^n$

```
rsolve(a(n) + a(n-1) - 6*a(n-2), a(n), {a(0): -1, a(1): 8})
```

b) $a_n = 2^n\left(\frac{n}{2} + 1\right)$

```
rsolve(-a(n+2) + 4*a(n+1) - 4*a(n), a(n), {a(0): 1, a(1):3})
```

c) $a_n = \frac{(-1)^n}{6} + \frac{5}{2} - \frac{8 \cdot 2^{-n}}{3}$

```
rsolve( -2*a(n+3) + a(n+2) + 2*a(n+1) - a(n) ,
        a(n) ,
        {a(0): 1, a(1): 1, a(2): 2}
      )
```

□

Ví dụ 9.11. Tìm công thức tường minh của dãy số Fibonacci, có hệ thức đệ quy $F_{n+2} = F_{n+1} + F_n,\ \forall n \geq 0$, trong đó $F_0 = 0, F_1 = 1$.

Giải. $F_n = \frac{1}{\sqrt{5}}\left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right],\ n \geq 0.$ □

Đặt $\alpha = \frac{1+\sqrt{5}}{2}$, $\beta = \frac{1-\sqrt{5}}{2}$, α gọi là *tỷ lệ vàng*, ta được *dạng Binet*[*] của F_n

$$F_n = \frac{1}{\sqrt{5}}(\alpha^n - \beta^n) = \frac{\alpha^n - \beta^n}{\alpha - \beta}.$$

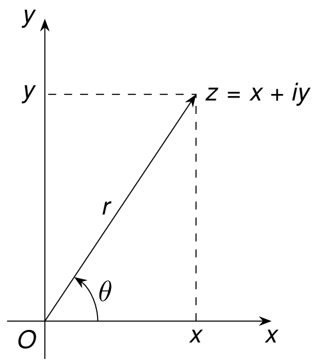
Nếu công thức nghiệm xuất hiện số phức dạng z^n, với $z = x + iy$, $n \in \mathbb{N}$, ta đưa z về dạng lượng giác

$$z = re^{i\theta} = r(\cos\theta + i\sin\theta),$$

trong đó $r = \sqrt{x^2 + y^2}$, và $\cos\theta = \frac{x}{r}$, $\sin\theta = \frac{y}{r}$. Khi đó, theo định lý DeMoivre

$$z^n = (re^{i\theta})^n = r^n e^{in\theta} = r^n(\cos n\theta + i\sin n\theta).$$

[*] do Jacques Philippe Marie Binet (1786–1856) phát biểu năm 1843



Ví dụ 9.12. Giải hệ thức $a_n = 2(a_{n-1} - a_{n-2})$, $\forall n \geq 2$, trong đó $a_0 = 1$, $a_1 = 2$.

Giải. $u_n = \dfrac{(1-i)^n(1+i)}{2} + \dfrac{(1-i)(1+i)^n}{2}$

```
n = symbols('n', integer=True)    # biến nguyên bất định n
rsolve( a(n)-2*(a(n-1)-a(n-2)) , a(n) , {a(0):1, a(1):2} ).
    simplify()
```

Biến đổi về dạng số thực, ta có $u_n = 2^{\frac{n}{2}}\left(\cos\dfrac{n\pi}{4} + \sin\dfrac{n\pi}{4}\right)$.

```
polar = lambda z: abs(z) * E**(I*arg(z))
re( polar(1-I)**n *(1+I)/2 + (1-I)*polar(1+I)**n /2 )
```

□

Ví dụ 9.13. Với $n \geq 0$, đặt $S = \{1, 2, 3, \ldots, n\}$ (khi $n = 0$, $S = \varnothing$), và a_n là số tập con của S không có các số nguyên liên tiếp. Tìm và giải hệ thức đệ quy của a_n.

a) Với $0 \leq n \leq 4$, liệt kê các tập con của S không có các số nguyên liên tiếp. Từ đó xác định a_n.

b) Lập hệ thức đệ quy và giải a_n.

Giải. a)

n	Tập con thỏa mãn giả thiết	a_n
0	$\varnothing$	1
1	$\varnothing, \{1\}$	2
2	$\varnothing, \{1\}, \{2\}$	3
3	$\varnothing, \{1\}, \{2\}, \{3\}, \{1,3\}$	5
4	$\{4\}, \{1,4\}, \{2,4\}$, và $\varnothing, \{1\}, \{2\}, \{3\}, \{1,3\}$	8

b) Giả sử, với $n \geq 2$, xét tập $A \subseteq S$ không có các số nguyên liên tiếp. Có hai khả năng:

1) $n \in A$: khi đó $n - 1 \notin A$, và $A - \{n\} \subseteq \{1, 2, \dots, n-2\}$ không có các số nguyên liên tiếp. Theo định nghĩa, có a_{n-2} tập như vậy.

2) $n \notin A$: lúc này $A \subseteq \{1, 2, \dots, n-1\}$ không có các số nguyên liên tiếp, và có a_{n-1} tập như vậy.

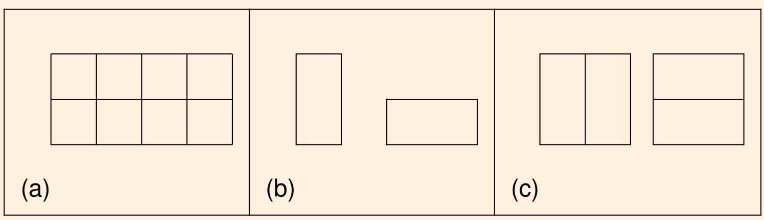
Do đó, theo quy tắc cộng, $a_n = a_{n-1} + a_{n-2}$, $\forall n \geq 2$, trong đó $a_0 = 1, a_1 = 2$. Giải hệ thức này, được

$$a_n = \frac{5 + 3\sqrt{5}}{10}\Big(\frac{1+\sqrt{5}}{2}\Big)^n - \frac{3\sqrt{5} - 5}{10}\Big(\frac{1-\sqrt{5}}{2}\Big)^n.$$

□

Ta thấy $a_n = F_{n+2} = \frac{1}{\sqrt{5}}\Big[\Big(\frac{1+\sqrt{5}}{2}\Big)^{n+2} - \Big(\frac{1-\sqrt{5}}{2}\Big)^{n+2}\Big]$.

Ví dụ 9.14. Xét bàn cờ $2 \times n$, với $n \in \mathbb{Z}^+$. Trường hợp $n = 4$ được cho trong phần (a) của hình dưới. Ta muốn phủ bàn cờ này bằng các hình domino dọc cỡ 2×1 và ngang cỡ 1×2. Các domino (hoặc viên gạch) này được cho trong phần (b) của hình.



Đặt a_n là số cách phủ (hay lợp) bàn cờ $2 \times n$ bằng các domino 2×1 và 1×2. Ở đây $a_1 = 1$, vì bàn cờ 2×1 chỉ có cách phủ bằng một domino dọc. Bàn cờ 2×2 có thể phủ bằng hai cách: dùng hai domino dọc, hoặc hai domino ngang, như phần (c) của hình. Như vậy $a_2 = 2$.
Lập hệ thức đệ quy và giải a_n.

Giải. Với $n \geq 3$, xét cột đầu tiên của bàn cờ $2 \times n$. Có thể phủ cột này theo hai cách.

1) Bằng một domino dọc: phần còn lại, là bàn cờ $2 \times (n-1)$, có a_{n-1} cách phủ.

2) Bằng hai domino ngang để phủ cả hai cột đầu bên trái: phần còn lại, là bàn cờ $2 \times (n-2)$, có a_{n-2} cách phủ.

Theo quy tắc cộng, $a_n = a_{n-1} + a_{n-2}$, $\forall n \geq 3$, trong đó $a_1 = 1, a_2 = 2$. Giải hệ thức này, được

$$a_n = \Big(\frac{1}{2} - \frac{\sqrt{5}}{2}\Big)^n\Big(\frac{1}{2} - \frac{\sqrt{5}}{10}\Big) + \Big(\frac{1}{2} + \frac{\sqrt{5}}{2}\Big)^n\Big(\frac{\sqrt{5}}{10} + \frac{1}{2}\Big)$$

$$= \Big(\frac{1+\sqrt{5}}{2}\Big)^n\Big(\frac{1}{2\sqrt{5}}+\frac{1}{2}\Big) - \Big(\frac{1-\sqrt{5}}{2}\Big)^n\Big(\frac{1}{2\sqrt{5}}-\frac{1}{2}\Big)$$
$$= \frac{1}{\sqrt{5}}\Big[\Big(\frac{1+\sqrt{5}}{2}\Big)^{n+1} - \Big(\frac{1-\sqrt{5}}{2}\Big)^{n+1}\Big],$$

□

Trong ví dụ trên $a_n = F_{n+1}$.

Sử dụng tính chất của số Fibonacci [có thể chứng minh bằng nguyên lý quy nạp],

$$F_n > \alpha^{n-2},\ \forall n \geq 3,\ \text{với } \alpha = \frac{1+\sqrt{5}}{2},$$

Gabriel Lamé* đã chứng minh

Ví dụ 9.15 (Định lý Lamé)**.** Cho $a, b \in \mathbb{Z}^+$, $a, b \geq 2$. Số phép chia dùng trong thuật toán Euclid để tìm ước chung lớn nhất của a và b không quá 5 lần số chữ số của b.

Giải. Đặt $r_0 = a$ và $r_1 = b$, ta có

$$\begin{aligned}
r_0 &= r_1q_1 + r_2, && 0 < r_2 < r_1\\
r_1 &= r_2q_2 + r_3, && 0 < r_3 < r_2\\
r_2 &= r_3q_3 + r_4, && 0 < r_4 < r_3\\
&\dots\dots\dots\dots\dots\dots\\
r_{n-2} &= r_{n-1}q_{n-1} + r_n, && 0 < r_n < r_{n-1}\\
r_{n-1} &= r_nq_n.
\end{aligned}$$

Khi đó, $\gcd(a, b) = r_n$, là phần dư khác không cuối cùng, và thuật toán thực hiện n phép chia.

Ta thấy, $q_i \geq 1,\ \forall i = \overline{1, n}$. Riêng $q_n \geq 2$, vì $r_{n-1} = r_nq_n$ mà $0 < r_n < r_{n-1}$. Như vậy

$$\begin{aligned}
r_n > 0 \Rightarrow r_n &\geq 1 = F_2\\
r_{n-1} = r_nq_n &\geq 1 \cdot 2 = 2 = F_3\\
r_{n-2} = r_{n-1}q_{n-1} + r_n &\geq F_3 \cdot 1 + F_2 = F_4\\
r_{n-3} = r_{n-2}q_{n-2} + r_{n-1} &\geq F_4 \cdot 1 + F_3 = F_5\\
&\dots\dots\dots\dots\dots\dots\\
r_2 = r_3q_3 + r_4 &\geq F_{n-1} \cdot 1 + F_{n-2} = F_n\\
b = r_1 = r_2q_2 + r_3 &\geq F_n \cdot 1 + F_{n-1} = F_{n+1}
\end{aligned}$$

*Gabriel Lamé, 1795–1870, nhà toán học Pháp

Dẫn đến

$$b \geq F_{n+1} > \alpha^{(n+1)-2} = \alpha^{n-1}$$
$$\Rightarrow n-1 < \log_\alpha b = \log_\alpha 10 \cdot \log_{10} b = 4.784971 \log_{10} b < 5\log_{10} b.$$

Nếu b có k chữ số, thì $10^{k-1} \leq b < 10^k$, nên $\log_{10} b < k$. Do đó $n-1 < 5k$, hay $n \leq 5k$, tức là số phép chia trong thuật toán Euclid không quá 5 lần số chữ số của b. □

Ví dụ 9.16. Tìm hệ thức đệ quy của a_n, là số xâu nhị phân độ dài n không có các số 0 liên tiếp.

Giải. **Cách 1:** Với mỗi xâu đếm bởi a_n, có hai khả năng:

1) Số đầu là 1, thì phần còn lại là xâu độ dài $n-1$ không có các số 0 liên tiếp. Số xâu như vậy là a_{n-1}.
2) Số đầu là 0, thì số thứ hai phải là 1, và phần còn lại là xâu độ dài $n-2$ không có các số 0 liên tiếp. Số các xâu như vậy là a_{n-2}.

Theo quy tắc cộng, $a_n = a_{n-1} + a_{n-2},\ \forall n \geq 3$. Ta xác định thêm $a_1 = 2$, $a_2 = 3$.

Cách 2: Cách này sử dụng các biến phụ. Trong các xâu đếm bởi a_n, đặt $a_n^{(0)}$ là số xâu số đầu là 0, và $a_n^{(1)}$ là số xâu có số đầu là 1. Khi đó $a_n = a_n^{(0)} + a_n^{(1)}$. Vì mỗi xâu dạng $1s$ đếm bởi $a_n^{(1)}$ khi và chỉ khi xâu s đếm bởi a_{n-1}, nên $a_n^{(1)} = a_{n-1}$.

Với mỗi xâu đếm bởi a_n, có hai khả năng:

1) Số thứ hai là 0, thì số đầu chỉ có thể là 1. Số xâu như vậy là $a_{n-1}^{(0)}$.
2) Số thứ hai là 1, thì số đầu có hai lựa chọn, là 0 hoặc 1. Số xâu như vậy là $2a_{n-1}^{(1)}$.

Theo quy tắc cộng

$$a_n = a_{n-1}^{(0)} + 2a_{n-1}^{(1)} = [a_{n-1}^{(0)} + a_{n-1}^{(1)}] + a_{n-1}^{(1)} = a_{n-1} + a_{n-2},\ \forall n \geq 3.$$

□

Ví dụ 9.17. Trong ngôn ngữ lập trình, xét các biểu thức số học hợp lệ, không có dấu ngoặc, gồm các chữ số 0, 1, 2,..., 9 và các phép toán hai ngôi $+, *, /$. Chẳng hạn, $3+4$ và $2+3*5$ là biểu thức số học hợp lệ; còn $8+*9$ thì không. Ở đây $2+3*5 = 17$, vì có thứ tự ưu tiên phép toán: phép nhân và chia thực hiện trước phép cộng, các phép toán cùng cấp thực

hiện từ trái sang phải.
Với $n \in \mathbb{Z}^+$, đặt a_n là số biểu thức số học hợp lệ gồm n ký tự.

a) Xác định a_1, a_2.

b) Lập hệ thức đệ quy và giải a_n.

Giải. a) $a_1 = 10$, vì biểu thức hợp lệ có một ký tự sẽ là một trong 10 chữ số. Tiếp theo $a_2 = 100$, gồm các biểu thức $00, 01, \ldots, 09, 10, 11, \ldots, 99$ (không cần dấu cộng ở đầu).

b) Khi $n \geq 3$, với mỗi biểu thức đếm bởi a_n, có hai khả năng:

1) $n - 1$ ký tự đầu cũng là biểu thức hợp lệ, thì ký tự cuối phải là một chữ số. Số biểu thức như vậy là $10a_{n-1}$.

2) $n - 1$ ký tự đầu không là biểu thức hợp lệ. Khi đó, ký tự thứ $n - 1$ là phép toán, ký tự cuối là chữ số, và $n - 2$ ký tự đầu là biểu thức hợp lệ. Vì nhóm hai ký tự cuối có 29 lựa chọn, gồm $+0, +1, \ldots, +9$, $*0, *1, \ldots, *9$, và $/1, /2, \ldots, /9$, nên trường hợp này có $29a_{n-2}$ biểu thức.

Theo quy tắc cộng, $a_n = 10a_{n-1} + 29a_{n-2}$, $\forall n \geq 3$. Giải hệ thức truy hồi này với $a_1 = 10$ và $a_2 = 100$, được

$$a_n = \frac{5}{3\sqrt{6}}[(5 + 3\sqrt{6})^n - (5 - 3\sqrt{6})^n].$$

□

Trong Ví dụ 8.12, ta đã đếm các tổng riêng đối xứng của n, tức là, số cách viết n thành dãy các số nguyên dương có tổng bằng n, mà cách đọc phép toán từ trái sang phải hay từ phải sang trái là như nhau. Đó là $a_n = 2^{\lfloor n/2 \rfloor}$.

Ví dụ 9.18. a) Lập hệ thức đệ quy và giải a_n, là số các tổng riêng đối xứng của số nguyên dương n.

b) Mô tả cách xây dựng các tổng riêng đối xứng của $n = 5, 6$ theo tổng riêng của các số nhỏ hơn.

Giải. a) Với mỗi tổng riêng đối xứng của n, có hai khả năng:

1) Số đầu và số cuối lớn hơn 1. Khi đó, nếu trừ hai số này đi 1, ta được tổng riêng đối xứng của $n - 2$. Số tổng riêng như vậy là a_{n-2}.

2) Số đầu và số cuối bằng 1. Khi đó, nếu bỏ đi hai số này, ta lại được tổng riêng đối xứng của $n - 2$. Số tổng riêng như vậy là a_{n-2}.

Theo quy tắc cộng, $a_n = a_{n-2} + a_{n-2} = 2a_{n-2}$, $\forall n \geq 3$. Ta xác định thêm $a_1 = 1$, $a_2 = 2$, và suy ra

$$a_n = \left(\frac{1}{2} + \frac{\sqrt{2}}{4}\right)(\sqrt{2})^n + \left(\frac{1}{2} - \frac{\sqrt{2}}{4}\right)(-\sqrt{2})^n.$$

Khi $n = 2k$ hoặc $n = 2k + 1$, ta đều có $k = \lfloor n/2 \rfloor$, và $a_n = 2^k = 2^{\lfloor n/2 \rfloor}$.

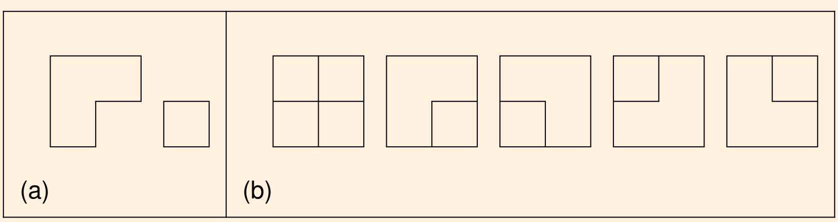
```
n,k = symbols('n k', integer=True)
sol = rsolve( a(n) - 2*a(n-2) , a(n) , {a(1):1, a(2):2})
sol
sol.subs(n, 2*k).simplify()
sol.subs(n, 2*k+1).simplify()
```

b) Theo lập luận ở ý (a), ta xây dựng tổng riêng của $n = 5, 6$ từ tổng riêng của 3, 4, tương ứng.

(1)	3	(1')	5	(1)	4	(1')	6
(2)	1 + 1 + 1	(2')	2 + 1 + 2	(2)	1 + 2 + 1	(2')	2 + 2 + 2
		(1")	1 + 3 + 1	(3)	2 + 2	(3')	3 + 3
		(2")	1 + 1 + 1 + 1 + 1	(4)	1 + 1 + 1 + 1	(4')	2 + 1 + 1 + 2
						(1")	1 + 4 + 1
						(2")	1 + 1 + 2 + 1 + 1
						(3")	1 + 2 + 2 + 1
						(4")	1 + 1 + 1 + 1 + 1 + 1

□

Ví dụ 9.19. Với $n \geq 1$, đặt a_n là số cách lát bàn cờ $2 \times n$ bằng hai loại gạch trong phần (a) của hình dưới. Ta thấy $a_1 = 1$, vì chỉ có một cách lát bàn cờ 2×1 bằng hai viên 1×1. Phần (b) của hình cho ta $a_2 = 5$. Còn bàn cờ 2×3 có 11 cách lát gồm: (1) một cách dùng 6 viên 1×1; (2) $4 \cdot 2 = 8$ cách dùng một viên lớn và ba viên 1×1; và (3) hai cách dùng cả hai viên lớn.



Lập hệ thức đệ quy và giải a_n.

Giải. Với mỗi cách lát bàn cờ $2 \times n$, xét ba khả năng lát cột 1:

a) Cột 1 được lát bằng hai viên 1×1, thì các cột còn lại, tức là bàn cờ $2 \times (n-1)$, có a_{n-1} cách lát.

b) Cột 1 chỉ có một viên 1×1, thì khi lát tiếp bằng viên lớn, sẽ lát hết hai cột đầu. Có hai cách lát như vậy cho hai cột đầu. Sau đó, $n-2$ cột còn lại có a_{n-2} cách lát, nên trường hợp này có $2a_{n-2}$ cách lát.

c) Cột 1 không có viên 1×1 nào. Xét hai khả năng lát cột 2:

   1) Cột 2 có một viên 1×1, thì có hai cột đầu được lát hết, và bằng hai cách. Sau đó, $n-2$ cột còn lại có a_{n-2} cách lát, nên trường hợp này có $2a_{n-2}$ cách lát.

   2) Cột 2 được lát tiếp bằng viên lớn, thì ba cột đầu được lát hết, và bằng hai cách. Sau đó, $n-3$ cột còn lại có a_{n-3} cách lát, nên trường hợp này có $2a_{n-3}$ cách lát.

Theo quy tắc cộng, $a_n = a_{n-1} + 2a_{n-2} + 2a_{n-2} + 2a_{n-3} = a_{n-1} + 4a_{n-2} + 2a_{n-3}$, $\forall n \geq 4$, trong đó $a_1 = 1$, $a_2 = 5$, $a_3 = 11$. Do đó

$$a_n = (-1)^n - \frac{1}{\sqrt{3}}[(1+\sqrt{3})^n - (1-\sqrt{3})^n].$$

□

Ví dụ 9.20. Cho $a \in \mathbb{R}^*$, xét định thức cấp n

$$D_n = \begin{vmatrix} a & a & 0 & 0 & \dots & 0 & 0 & 0 \\ a & a & a & 0 & \dots & 0 & 0 & 0 \\ 0 & a & a & a & \dots & 0 & 0 & 0 \\ 0 & 0 & a & a & \dots & 0 & 0 & 0 \\ & & \dots & \dots & \dots & \dots & & \\ 0 & 0 & 0 & 0 & \dots & a & a & 0 \\ 0 & 0 & 0 & 0 & \dots & a & a & a \\ 0 & 0 & 0 & 0 & \dots & 0 & a & a \end{vmatrix}_n$$

Tìm công thức của D_n chỉ phụ thuộc n.

Giải. Khai triển D_n theo hàng đầu

$$D_n = a\underbrace{\begin{vmatrix} a & a & 0 & 0 & \dots & 0 & 0 & 0 \\ a & a & a & 0 & \dots & 0 & 0 & 0 \\ 0 & a & a & a & \dots & 0 & 0 & 0 \\ 0 & 0 & a & a & \dots & 0 & 0 & 0 \\ & & \dots & \dots & \dots & \dots & & \\ 0 & 0 & 0 & 0 & \dots & a & a & 0 \\ 0 & 0 & 0 & 0 & \dots & a & a & a \\ 0 & 0 & 0 & 0 & \dots & 0 & a & a \end{vmatrix}_{n-1}}_{D_{n-1}} - a\begin{vmatrix} a & a & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & a & a & 0 & \dots & 0 & 0 & 0 \\ 0 & a & a & a & \dots & 0 & 0 & 0 \\ 0 & 0 & a & a & \dots & 0 & 0 & 0 \\ & & \dots & \dots & \dots & \dots & & \\ 0 & 0 & 0 & 0 & \dots & a & a & 0 \\ 0 & 0 & 0 & 0 & \dots & a & a & a \\ 0 & 0 & 0 & 0 & \dots & 0 & a & a \end{vmatrix}_{n-1}$$

Khai triển định thức thứ hai theo cột đầu, được

$$D_n = aD_{n-1} - a(aD_{n-2}) = aD_{n-1} - a^2 D_{n-2},\ \forall n \geq 3,$$

trong đó $D_1 = \left|[a]\right| = a$, $D_2 = \begin{vmatrix} a & a \\ a & a \end{vmatrix} = 0$. Suy ra

$$\begin{aligned} D_n &= \left(\frac{a(1-\sqrt{3}i)}{2}\right)^n\left(\frac{1}{2} + \frac{\sqrt{3}i}{6}\right) + \left(\frac{a\left(1+\sqrt{3}i\right)}{2}\right)^n\left(\frac{1}{2} - \frac{\sqrt{3}i}{6}\right) \\ &= \frac{2}{\sqrt{3}}a^n \sin\frac{(n+1)\pi}{3}. \end{aligned}$$

□

```
1 n = symbols('n', integer=True)
2 Matrix([[a]]).det()
3 Matrix([[a, a], [a, a]]).det()
4 rsolve(-D(n) + a*D(n-1) - a**2 * D(n-2), D(n), {D(1):a, D(2):0})
5 print(_)

6 polar = lambda z: abs(z) * E**(I*arg(z))
7 # Từ kết quả của dòng 5: đưa z trong z^n về dạng lượng giác, viết số thực 1/2 ở
    dạng phân số
8 re( (polar(1 - sqrt(3)*I)/2)**n *(Rational(1,2) + sqrt(3)*I/6) +
    (polar(1 + sqrt(3)*I)/2)**n* (Rational(1,2) - sqrt(3)*I/6) ).
    simplify()
```

Bài tập 9.2

9.6. Giải các hệ thức đệ quy

a) $a_n = 5a_{n-1} + 6a_{n-2}$; $a_0 = 1$, $a_1 = 3$

b) $2a_{n+2} - 11a_{n+1} + 5a_n = 0$; $a_0 = 2$, $a_1 = -8$

c) $a_{n+2} + a_n = 0$; $a_0 = 0$, $a_1 = 3$

d) $a_n - 6a_{n-1} + 9a_{n-2} = 0$; $a_0 = 5$, $a_1 = 12$

e) $a_n + 2a_{n-1} + 2a_{n-1} = 0$; $a_0 = 1$, $a_1 = 3$

9.7. Xét quan hệ đệ quy $a_{n+2} + ba_{n+1} + ca_n = 0$. Biết $a_0 = 0$, $a_1 = 1$, $a_2 = 4$, và $a_3 = 37$, tìm b, c và giải hệ thức đệ quy.

9.8. Giải hệ thức đệ quy $a_{n+2} = a_{n+1}a_n$; $a_0 = 1$, $a_1 = 2$.

9.9. Tìm và giải hệ thức đệ quy của $D_n = \begin{vmatrix} 2 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 2 & 1 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & \dots & 0 & 0 & 0 \\ & & \dots & \dots & \dots & \dots & & \\ 0 & 0 & 0 & 0 & \dots & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & \dots & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & 2 \end{vmatrix}_n$

9.10. Giải hệ thức đệ quy $a_{n+2}^2 - 5a_{n+1}^2 + 4a_n^2 = 0$; $a_0 = 4$, $a_1 = 13$.

9.3 Hệ thức đệ quy không thuần nhất

Ví dụ 9.21. Với $n \geq 1$, đặt $X_n = \{1, 2, \dots, n\}$, $\mathcal{P}(X_n)$ là tập lũy thừa của X_n. Gọi a_n là số cạnh trong biểu đồ Hasse của quan hệ thứ tự $(\mathcal{P}(X_n), \subseteq)$.
Lập hệ thức đệ quy và giải a_n.

Giải. □

Bài tập 9.3

9.11. Giải các hệ thức đệ quy

a) $a_{n+1} - a_n = 2n + 3$, $a_0 = 1$

b) $a_{n+1} - a_n = 3n^2 - n$, $a_0 = 3$

c) $a_{n+1} - 2a_n = 5$, $a_0 = 1$

d) $a_{n+1} - 2a_n = 2^n$, $a_0 = 1$

9.12. Lập quan hệ đệ quy cho tổng $a_n = \sum_{i=0}^{n} i^2$.

9.13. Giải các hệ thức đệ quy

a) $a_{n+2} + 3a_{n+1} + 2a_n = 3^n$; $a_0 = 0$, $a_1 = 1$

b) $a_{n+2} + 4a_{n+1} + 4a_n = 7$; $a_0 = 1$, $a_1 = 2$

9.14. Giải hệ thức đệ quy $a_{n+2} - 6a_{n+1} + 9a_n = 3 \cdot 2^n + 7 \cdot 3^n$; $a_0 = 1$, $a_1 = 4$

9.15. Tìm nghiệm tổng quát của quan hệ đệ quy $a_{n+3} - 3a_{n+2} + 3a_{n+1} - a_n = 3 + 5n$.

9.16. Nghiệm tổng quát của quan hệ đệ quy $a_{n+2} + b_1 a_{n+1} + b_2 a_n = b_3 n + b_4$, với các hằng số b_i, $1 \leq i \leq 4$, là $c_1 2^2 + c_2 3^n + n - 7$. Tìm b_i, $1 \leq i \leq 4$.

9.17. Giải các quan hệ đệ quy

a) $a_{n+2}^2 - 5a_{n+1}^2 + 6a_n^2 = 7n$, $a_0 = a_1 = 1$

b) $a_n^2 - 2a_{n-1} = 0$, $a_0 = 2$. [*Gợi ý:* đặt $b_n = \log_2 a_n$]

9.4 Phương pháp hàm sinh

Giả sử dãy a_n có quan hệ đệ quy. Xét hàm sinh $f(x) = \sum_{n=0}^{\infty} a_n x^n$. Căn cứ vào quan hệ đệ quy của a_n, bằng phép biến đổi phù hợp, ta biểu diễn được $f(x)$ theo theo x và chính $f(x)$. Từ đó, ta giải được $f(x)$ là một hàm sơ cấp. Khi đó, a_n là hệ số của x^n trong khai triển MacLaurin của hàm này.

Ví dụ 9.22. Giải hệ thức đệ quy $a_n - 3a_{n-1} = n$, trong đó $a_0 = 1$.

Giải. Xét hàm sinh $f(x) = \sum_{n=0}^{\infty} a_n x^n = a_0 + \sum_{n=1}^{\infty} a_n x^n$. Thay $a_n = 3a_{n-1} + n$, với $n \geq 1$, và $a_0 = 1$:

$$f(x) = 1 + \sum_{n=1}^{\infty} (3a_{n-1} + n)x^n = 1 + \sum_{n=1}^{\infty} 3a_{n-1}x^n + \sum_{n=1}^{\infty} nx^n.$$

Trong ví dụ 8.14, ta đã tính được $\sum_{n=1}^{\infty} nx^n = \sum_{n=0}^{\infty} nx^n = \dfrac{x}{(1-x)^2}$. Suy ra

$$\begin{aligned} f(x) &= 1 + 3x \sum_{n=1}^{\infty} a_{n-1}x^{n-1} + \frac{x}{(1-x)^2} = 1 + 3x \sum_{n=0}^{\infty} a_n x^n + \frac{x}{(1-x)^2} \\ &= 1 + 3x \cdot f(x) + \frac{x}{(1-x)^2}. \end{aligned}$$

Giải phương trình bậc nhất với ẩn là $f(x)$, được phân thức

$$f(x) = -\frac{x + (x-1)^2}{(x-1)^2(3x-1)}.$$

Phân tích $f(x)$ thành tổng các phân thức đơn giản

$$f(x) = -\frac{7}{4(3x-1)} + \frac{1}{4(x-1)} - \frac{1}{2(x-1)^2}.$$

Viết các số hạng dưới dạng $(1+x)^n$:

$$f(x) = \frac{7}{4}(1-3x)^{-1} - \frac{1}{4}(1-x)^{-1} - \frac{1}{2}(1-x)^{-2}.$$

Vì a_n là hệ số của x_n trong khai triển MacLaurin của $f(x)$, nên

$$a_n = \frac{7}{4}\binom{-1}{n}(-3)^n - \frac{1}{4}\binom{-1}{n}(-1)^n - \frac{1}{2}\binom{-2}{n}(-1)^n.$$

Theo ví dụ 8.2, $\binom{-1}{n} = (-1)^n$, $\binom{-2}{n} = (-1)^n\binom{2+n-1}{n} = (-1)^n(n+1)$, suy ra

$$\begin{aligned} a_n &= \frac{7}{4}(-1)^n(-3)^n - \frac{1}{4}(-1)^n(-1)^n - \frac{1}{2}(-1)^n(n+1)(-1)^n \\ &= \frac{7}{4}3^n - \frac{1}{4} - \frac{n+1}{2} = \frac{7\cdot 3^n - 2n - 3}{4}. \end{aligned}$$

□

```
Sum( n * x**n, (n, 1, oo) ).doit()

y = symbols('y')  # đại diện cho f(x)
solve(-y + 1 + 3*x * y + x / (1-x)**2, y)
sol = _[0]
sol

sol.apart()
```

Ví dụ 9.23. Giải hệ thức đệ quy $a_{n+2} - 5a_{n+1} + 6a_n = 2$, trong đó $a_0 = 3$, $a_1 = 7$.

Giải. Xét

$$\begin{aligned} f(x) &= \sum_{n=0}^{\infty} a_n x^n = \sum_{n=2}^{\infty} a_n x^n + a_0 + a_1 x = \sum_{n=0}^{\infty} a_{n+2}x^{n+2} + 3 + 7x \\ &= \sum_{n=0}^{\infty} (5a_{n+1} - 6a_n + 2)x^{n+2} + 3 + 7x \\ &= \sum_{n=0}^{\infty} 5a_{n+1}x^{n+2} - \sum_{n=0}^{\infty} 6a_n x^{n+2} + \sum_{n=0}^{\infty} 2x^{n+2} + 3 + 7x \end{aligned}$$

$$= 5x\sum_{n=0}^{\infty} a_{n+1}x^{n+1} - 6x^2\sum_{n=0}^{\infty} a_n x^n + 2x^2\sum_{n=0}^{\infty} x^n + 3 + 7x$$
$$= 5x[f(x) - a_0] - 6x^2 f(x) + 2x^2\frac{1}{1-x} + 3 + 7x$$

Suy ra

$$f(x) = \frac{3-5x}{3x^2-4x+1} = -\frac{2}{3x-1} - \frac{1}{x-1} = 2(1-3x)^{-1} + (1-x)^{-1}$$

Do đó, hệ số của x^n trong khai triển Maclaurin của $f(x)$ là

$$a_n = 2\binom{-1}{n}(-3)^n + \binom{-1}{n}(-1)^n = 2(-1)^n(-3)^n + (-1)^n(-1)^n = 2\cdot 3^n + 1.$$

□

Ví dụ thứ ba, một kết quả quen thuộc về tổ hợp lặp.

Ví dụ 9.24. Với $n, r \in \mathbb{N}$, đặt $c(n, r)$ là số cách chọn r vật, có lặp, từ n vật. Chứng minh

a) $c(n, r)$ có hệ thức đệ quy $c(n, r) = c(n-1, r) + c(n, r-1)$, $\forall n, r \geq 1$.

b) $c(n, r) = \binom{n+r-1}{r}$.

Giải. a) Với $n \geq 1$, đánh nhãn các vật là 1, 2,. . . , n. Chỉ có hai khả năng:

1) Vật 1 không được chọn. Khi đó r vật được chọn từ $n-1$ kia. Ta có $c(n-1, r)$ cách.

2) Vật 1 được chọn ít nhất một lần. Khi đó ta cần chọn $r-1$ vật từ các vật 1, 2,. . . , n, rồi chọn tiếp một vật nhãn là 1 nữa. Ta có $c(n, r-1)$ cách.

Theo quy tắc cộng, $c(n, r) = c(n-1, r) + c(n, r-1)$, $\forall n, r \geq 1$. Ngoài ra $c(n, 0) = 1$, $\forall n \geq 0$ và $c(0, r) = 0$, $\forall r > 0$.

b) Đặt $f_n = \sum_{r=0}^{\infty} c(n, r)x^r$ là hàm sinh dãy $c(n, 0), c(n, 1), c(n, 2)$, ... [Ở đây ta viết tắt $f_n(x)$ là f_n.] Ta có

$$f_n = c(n, 0) + \sum_{r=1}^{\infty} c(n, r)x^r$$
$$= 1 + \sum_{r=1}^{\infty} [c(n-1, r) + c(n, r-1)]x^r$$

$$= 1 + \sum_{r=1}^{\infty} c(n-1, r)x^r + \sum_{r=1}^{\infty} c(n, r-1)x^r$$
$$= f_{n-1} + x\sum_{r=1}^{\infty} c(n, r-1)x^{r-1}$$
$$= f_{n-1} + xf_n.$$

Suy ra $f_n - xf_n = f_{n-1}$, hay $f_n = \dfrac{f_{n-1}}{1-x}$. Vì $f_0 = c(0,0) + c(0,1)x + c(0,2)x^2 + \cdots = 1 + 0 + 0 + \cdots = 1$, nên $f_n = \dfrac{1}{(1-x)^n} = (1-x)^{-n}$. Do đó $c(n, r)$ là hệ số của x^r trong $(1-x)^{-n}$, đó là $\binom{-n}{r}(-1)^r = \binom{n+r-1}{r}$.

□

Trong ví dụ cuối, ta dùng hàm sinh để giải hệ các hệ thức đệ quy.

Ví dụ 9.25.

$$a_{n+1} = 2a_n + b_n$$
$$b_{n+1} = a_n + b_n,$$

trong đó $a_0 = 1, b_0 = 0$.

Giải. Xét $f(x) = \sum_{n=0}^{\infty} a_n x^n$ và $g(x) = \sum_{n=0}^{\infty} b_n x^n$. Ta có

$$f(x) = \sum_{n=1}^{\infty} a_n x^n + a_0 = \sum_{n=0}^{\infty} a_{n+1}x^{n+1} + a_0 = \sum_{n=0}^{\infty}(2a_n + b_n)x^{n+1} + 1$$
$$= x\Big(2\sum_{n=0}^{\infty} a_n x^n + \sum_{n=0}^{\infty} b_n x^n\Big) + 1 = x[2f(x) + g(x)] + 1, \quad \text{và}$$
$$g(x) = \sum_{n=1}^{\infty} b_n x^n + b_0 = \sum_{n=0}^{\infty} b_{n+1}x^{n+1} + b_0 = \sum_{n=0}^{\infty}(a_n + b_n)x^{n+1} + 0$$
$$= x\Big(\sum_{n=0}^{\infty} a_n x^n + \sum_{n=0}^{\infty} b_n x^n\Big) = x[f(x) + g(x)]$$

Giải hệ hai phương trình tuyến tính với ẩn là $f(x)$ và $g(x)$:

$$f(x) = \frac{1-x}{x^2 - 3x + 1}, \quad \text{và} \quad g(x) = \frac{x}{x^2 - 3x + 1}.$$

Phân tích $f(x)$ thành tổng các phân thức đơn giản

$$f(x) = \frac{1-x}{x^2-3x+1} = -\frac{\frac{1}{2} - \frac{\sqrt{5}}{10}}{x - \frac{3}{2} + \frac{\sqrt{5}}{2}} - \frac{\frac{\sqrt{5}}{10} + \frac{1}{2}}{x - \frac{3}{2} - \frac{\sqrt{5}}{2}} = \frac{5-\sqrt{5}}{10}\frac{1}{x_1 - x} + \frac{5+\sqrt{5}}{10}\frac{1}{x_2 - x},$$

trong đó $x_{1,2} = \dfrac{3 \mp \sqrt{5}}{2}$, với $x_1 x_2 = 1$. Suy ra

$$f(x) = \frac{5-\sqrt{5}}{10x_1}\frac{1}{1-\frac{x}{x_1}} + \frac{5+\sqrt{5}}{10x_2}\frac{1}{1-\frac{x}{x_2}} = \frac{5-\sqrt{5}}{10}x_2\frac{1}{1-x_2x} + \frac{5+\sqrt{5}}{10}x_1\frac{1}{1-x_1x}.$$

Do đó

$$a_n = \frac{5+\sqrt{5}}{10}x_1x_1^n + \frac{5-\sqrt{5}}{10}x_2x_2^n = \frac{5+\sqrt{5}}{10}\left(\frac{3-\sqrt{5}}{2}\right)^{n+1} + \frac{5-\sqrt{5}}{10}\left(\frac{3+\sqrt{5}}{2}\right)^{n+1}.$$

Tương tự $b_n = -\dfrac{5+3\sqrt{5}}{10}\left(\dfrac{3-\sqrt{5}}{2}\right)^{n+1} - \dfrac{5-3\sqrt{5}}{10}\left(\dfrac{3+\sqrt{5}}{2}\right)^{n+1}$. □

```
solve([-y + x * (2*y + z) + 1, -z + x * (y + z)], [y, z])  # y = f(x), z = g(x)
sol  # kết quả {y: (1 - x)/(x**2 - 3*x + 1), z: x/(x**2 - 3*x + 1)}
sol[y]  # f(x) = (1 - x)/(x^2 - 3x + 1)
# Khi mẫu số của f(x) không có nghiệm hữu tỷ, lệnh tách f(x) thành tổng các phân thức đơn giản:
sol[y].apart(full=True).doit()
```

Bài tập 9.4

9.18. Giải các quan hệ đệ quy bằng phương pháp hàm sinh

a) $a_{n+1} - a_n = 3^n$, $a_0 = 1$

b) $a_{n+1} - a_n = n^2$, $a_0 = 1$

c) $a_{n+2} - 3a_{n+1} + 2a_n = 0$; $a_0 = 1$, $a_1 = 6$

d) $a_{n+2} - 2a_{n+1} + a_n = 2^n$; $a_0 = 1$, $a_1 = 2$

9.19. Cho n vật phân biệt. Đặt $a(n,r)$ là số cách chọn, không lặp, r vật từ n vật, với $0 \le r \le n$. Ở đây $a(n,r) = 0$ nếu $r > n$. Dùng quan hệ đệ quy $a(n,r) = a(n-1,r-1) + a(n-1,r)$, với $n, r \ge 1$ để chứng minh $f(x) = (1+x)^n$ sinh dãy $a(n,r)$, $r \ge 0$.

9.20. Giải hệ các hệ thức đệ quy

a) $a_{n+1} = -2a_n - 4b_n$, $b_{n+1} = 4a_n + 6b_n$; $a_0 = 1$, $b_0 = 0$

b) $a_{n+1} = 2a_n - b_n + 2$, $b_{n+1} = -a_n + 2b_n - 1$; $a_0 = 0$, $b_0 = 1$

9.5 Hệ thức đệ quy phi tuyến đặc biệt

Ví dụ 9.26.

Ví dụ 9.27.

Ví dụ 9.28.

9.6 Thuật toán chia để trị

Để giải một bài toán với đầu vào cỡ n:

1) Chia bài toán thành a bài toán nhỏ cùng dạng với bài toán ban đầu, nhưng cỡ là $\frac{n}{b}$, chính xác là $\left\lfloor \frac{n}{b} \right\rfloor$ hoặc $\left\lceil \frac{n}{b} \right\rceil$. Ở đây $a \geq 1, b \geq 2$ là các hằng số nguyên.
2) Tổng hợp kết của từ các bài toán nhỏ, để suy ra kết quả của bài toán ban đầu.

Gọi $f(n)$ là độ phức tạp của thuật toán với cỡ đầu vào n, $h(n)$ là độ phức tạp của bước (2). Khi đó

$$f(n) = af\left(\frac{n}{b}\right) + h(n).$$

Để đơn giản ta xét các giá trị của n là 1, b, b^2, b^3, ..., tức là $n = b^k$ với $k \in \mathbb{N}$. Khi đó

$$f(b^k) = af(b^{k-1}) + h(b^k).$$

Đặt $u_k = f(b^k)$, ta được hệ thức đệ quy tuyến tính không thuần nhất cấp một hệ số hằng

$$u_k = au_{k-1} + h(b^k).$$

trong đó $u_0 = f(1)$. Khi đó

$$f(n) = f(b^k) = u_k = u_{\log_b n}.$$

Một số thuật toán chia để trị điển hình như tìm số lớn nhất của dãy, tìm kiếm nhị phân, và sắp xếp trộn.

Ví dụ 9.29. Trình bày một thuật toán chia để trị để tìm số lớn nhất M của dãy $a_1, a_2, \dots, a_n$, và đánh giá độ phức tạp của thuật toán.

Giải. Xét thuật toán chia để trị sau:

1) Nếu dãy chỉ có một phần tử, thì $M = a_1$. Nếu không thì xuống thực hiện bước (2).
2) Chia đôi dãy được hai dãy con $a_1, a_2, \dots, a_k$ và $a_{k+1}, a_{k+2}, \dots, a_n$. Mỗi dãy có $\frac{n}{2}$ số. Gọi M_1, M_2 là số lớn nhất của hai dãy con này.
3) $M = \max\{M_1, M_2\}$.

Chia đôi dãy $a_1, a_2, \dots, a_n$ thành hai dãy con.

```
1  def my_max(a):
2    n = len(a)           # 1 phép tính, 1 phép gán
3    if n == 1:           # 1 phép so sánh
4      return a[0]
5    k = n // 2           # 1 phép toán
6    M1 = my_max(a[:k])   # bài toán với cỡ đầu vào n/2, và 1 phép gán
7    M2 = my_max(a[k:])   # như dòng 6
8    if M1 > M2:          # 1 phép so sánh
9      return M1
10   else:
11     return M2

12 my_max([2, 8, 6, 1, 0, 5, 4, 1, 7])  # → 8
```

Gọi $f(n)$ là số phép so sánh để tìm số lớn nhất của dãy cỡ n, được dùng để đánh giá độ phức tạp của thuật toán. Ta có

$$f(n) = \underbrace{1}_{\text{bước 1}} + \underbrace{2 \cdot f\left(\frac{n}{2}\right)}_{\text{bước 2}} + \underbrace{1}_{\text{bước 3}} = 2f\left(\frac{n}{2}\right) + 2.$$

Xét $n = 2^k$, $a_k = f(n) = f(2^k)$ thì

$$a_k = 2f(2^{k-1}) + 2 = 2a_{k-1} + 2$$

trong đó $a_0 = f(1) = 0$.

Giải hệ thức đệ quy, ta được $a_k = 3 \cdot 2^k - 2$, hay $f(n) = 3n - 2 \in O(n)$. Thuật toán có độ phức tạp tuyến tính.

Thay vì chỉ đếm số phép so sánh, nếu đặt $f(n)$ là bao gồm các phép toán, phép so sánh, và phép gán của thuật toán trên, thì $f(n) = 2f\left(\frac{n}{2}\right) + 7$, tương ứng $a_k = 2a_{k-1} + 7$, với $a_0 = f(1) = 3$. Khi đó $a_k = 5 \cdot 2^k - 2$, tức là $f(n) = 5n - 2$. □

Ví dụ 9.30 (Tìm kiếm nhị phân)**.** Tìm vị trí của x trong dãy đơn điệu $a_1 < a_2 < \dots < a_n$: nêu một thuật toán chia để trị và đánh giá độ phức tạp của thuật toán.

Giải. Xét thuật toán chia để trị tìm vị trí của x trong dãy $a_i, a_{i+1}, \dots, a_j$ với $i \leq j$. Ban đầu $i = 1, j = n$.

1) Nếu dãy chỉ có một phần tử, tức là $i = j$, thì chỉ cần so sánh x với a_i [Nếu $x = a_i$ thì i là vị trí của x trong dãy, ngược lại không tìm được x trong dãy]. Ngược lại thì xuống thực hiện bước (2).

2) Chia đôi dãy được hai dãy con $a_i, a_{i+1}, \dots, a_k$ và $a_{k+1}, a_{k+2}, \dots, a_j$. Mỗi dãy có $\frac{n}{2}$ số.

Do tính đơn điệu của dãy, nên nếu $x \leq a_k$ thì chỉ cần tìm x trong dãy thứ nhất, và nếu ngược lại, chỉ cần tìm x trong dãy thứ hai. Tức là, chỉ thực hiện một bài bài toán với cỡ đầu vào $\frac{n}{2}$.

```
1  def binary_search(x, a, i, j):
2    if i == j:            # 1 phép so sánh
3      if x == a[i]:       # 1 phép so sánh
4        return i
5      else:
6        return 'Không thấy'
7    k = (i + j) // 2      # 2 phép toán, 1 phép gán
8    if x <= a[k]:         # 1 phép so sánh
9      return binary_search(x, a, i, k)       # bài toán với cỡ đầu vào n/2
10   else:
11     return binary_search(x, a, k + 1, j)   # như dòng 9

12 a = [0, 2, 3, 6, 7, 10, 11, 15]
13 binary_search(10, a, 0, len(a) - 1)        # → 5
```

Đặt $f(n)$ là số phép so sánh cho bài toán với đầu vào cỡ n. Khi đó

$$f(n) = \underbrace{2}_{\text{dòng 2, 8}} + \underbrace{f\left(\frac{n}{2}\right)}_{\text{dòng 9 hoặc 11}}$$

Xét $n = 2^k \Leftrightarrow k = \log_2 n$, và $a_k = f(n) = f(2^k)$, thì

$$a_k = 2 + f(2^{k-1}) = 2 + a_{k-1}$$

trong đó $a_0 = f(1) = 2$ (dòng 2, 3). Suy ra $a_k = 2 + 2k$, tức là $f(n) = 2 + 2\log_2 n \in O(\log_2 n)$. Thuật toán có độ phức tạp loga. □

Ví dụ 9.31 (Sắp xếp trộn)**.** Sắp xếp dãy $a_1, a_2, \ldots, a_n$ thành dãy tăng dần: trình bày thuật toán chia để trị và đánh giá độ phức tạp của thuật toán.

Giải. Các bước của thuật toán:

1) Nếu dãy chỉ có 1 số, ta không cần làm gì cả. Ngược lại, chuyển xuống thực hiện bước (2).

2) Chia đôi dãy được hai dãy con $a_1, a_2, \ldots, a_k$ và $a_{k+1}, a_{k+2}, \ldots, a_n$ cỡ $\frac{n}{2}$. Sắp xếp hai dãy này, được hai dãy tăng.

3) Trộn hai dãy tăng này cho thành một dãy tăng.

Để thực hiện bước (3), ta dùng thuật toán dưới đây. Trong thuật toán này, để trộn hai dãy tăng cỡ m và n, số chu trình tối giản là $m + n$.

```
def merge(a, b):
  n, m = len(a), len(b)
  c = [0] * (n + m)
  i, j, k = 0, 0, 0
  while i < n and j < m:
    if a[i] < b[j]:
      c[k] = a[i]
      i += 1
    else:
      c[k] = b[j]
      j += 1
    k += 1

  while i < n:
    c[k] = a[i]
    i += 1
    k += 1
  while j < m:
    c[k] = b[j]
    j += 1
    k += 1
  return c


merge([2, 3], [1, 4, 5])   # → 1, 2, 3, 4, 5
```

Chương trình chính như sau:

```
1 def merge_sort(a):
2   n = len(a)
3   if n == 1:
4     return a
5   k = n // 2
6   L = merge_sort(a[:k])        # sắp xếp dãy cỡ n/2
7   R = merge_sort(a[k:])        # như dòng 6
8   return merge(L, R)           # n/2 + n/2 = n chu trình tối giản

9 merge_sort([1, 5, 3, 4, 2])  # → 1, 2, 3, 4, 5
```

Gọi $f(n)$ là số chu trình tối giản của thuật toán với dãy cỡ n. Ta có

$$f(n) = 2f\left(\frac{n}{2}\right) + n$$

Xét $n = 2^k \Leftrightarrow k = \log_2 n$, và $a_k = f(n) = f(2^k)$, thì

$$a_k = 2f(2^{k-1}) + 2^k = 2a_{k-1} + 2^k,$$

trong đó $a_0 = f(1) = 0$.

Giải hệ thức đệ quy này, được $a_k = 2^k k$, tức là $f(n) = n\log_2 n$. Thuật toán có độ phức tạp $n\log_2 n$.

□

Dưới đây là một số kết luận, đánh giá về hệ thức chia để trị.

Định lý 9.1. *Cho $a, b, c \in \mathbb{Z}^+$, $b \geq 2$, và $f : \mathbb{Z}^+ \to \mathbb{R}$. Nếu*

$$\begin{aligned} &f(1) = c, \quad \text{và} \\ &f(n) = af\Big(\frac{n}{b}\Big) + c, \quad \text{với } n = b^k, k \geq 1, \end{aligned}$$

thì với mọi $n = 1, b, b^2, \ldots$

$$f(n) = \begin{cases} c(\log_b n + 1) & \text{nếu } a = 1 \\ \dfrac{c(an^{\log_b a} - 1)}{a - 1} & \text{nếu } a \geq 2. \end{cases}$$

```
rsolve(-u(k) + a*u(k-1) + c, u(k), {u(0): c}).simplify()
rsolve(-u(k) +   u(k-1) + c, u(k), {u(0): c}).simplify()
```

Với giả thiết của định lý, ta thường đánh giá

$$f(n) \in \begin{cases} O(\log_b n)) & \text{nếu } a = 1 \\ O(n^{\log_b a}) & \text{nếu } a \geq 2. \end{cases}$$

Hệ quả 9.1. *Cho $a, b, c \in \mathbb{Z}^+$, $b \geq 2$, và $f : \mathbb{Z}^+ \to \mathbb{R}^+ \cup \{0\}$. Nếu*

$$\begin{aligned} &f(1) \leq c, \quad \text{và} \\ &f(n) \leq af\Big(\frac{n}{b}\Big) + c, \quad \text{với } n = b^k, k \geq 1, \end{aligned}$$

thì với mọi $n = 1, b, b^2, \ldots$

$$f(n) \in \begin{cases} O(\log_b n)) & \text{nếu } a = 1 \\ O(n^{\log_b a}) & \text{nếu } a \geq 2. \end{cases}$$

Bài tập 9.6

9.21. Cho $f : \mathbb{Z}^+ \to \mathbb{R}$. Giải quan hệ $f(n)$ trên tập S đã cho, và xác định dạng O–lớn của f trên S.

a) $f(1) = 5$; $f(n) = 4f\left(\frac{n}{3}\right) + 5$, $n = 3, 9, 27, \dots$; $S = \{3^i \mid i \in \mathbb{N}\}$

b) $f(1) = 7$; $f(n) = f\left(\frac{n}{5}\right) + 7$, $n = 5, 25, 125, \dots$; $S = \{5^i \mid i \in \mathbb{N}\}$

c) $f(1) = 0$; $f(n) = 2f\left(\frac{n}{5}\right) + 3$, $n = 5, 25, 125, \dots$; $S = \{5^i \mid i \in \mathbb{N}\}$

d) $f(1) = 1$; $f(n) = f\left(\frac{n}{2}\right) + 2$, $n = 2, 4, 8, \dots$; $S = \{2^i \mid i \in \mathbb{N}\}$

9.22. Cho $a, b, c \in \mathbb{Z}^+$, $b \geq 2$, và $d \in \mathbb{N}$. Giải hệ thức đệ quy

$$\begin{aligned} f(1) &= d \\ f(n) &= af\left(\frac{n}{b}\right) + c, \quad n = b^k, k \geq 1. \end{aligned}$$

Tóm tắt lệnh Python

- sympy
  - arg
  - det
  - E, I
  - Matrix
  - Rational
  - rsolve
  - solve

Bài tập bổ sung

Tài liệu tham khảo

[1] NumPy community. *NumPy User Guide*. phiên bản 1.22.4. 535 trang. URL: https://numpy.org/doc/stable.

[2] Judi J. McDonald David C. Lay Steven R. Lay. *Linear Algebra and Its Applications*. phiên bản 6. Pearson, 2022. 755 trang.

[3] Ralph P. Grimaldi. *Discrete and Combinatorial Mathematics: An Applied Introduction*. phiên bản 5. Pearson Addison-Wesley, 2004. 992 trang.

[4] Ralph P. Grimaldi. *Discrete and Combinatorial Mathematics: Instructor's Solutions Manual*. phiên bản 5. Pearson Addison-Wesley, 2004. 465 trang.

[5] Kenneth H. Rosen. *Discrete Mathematics and Its Applications*. phiên bản 8. McGraw-Hill Education, 2019. 1118 trang.

[6] Edward R. Scheinerman. *Mathematics: A Discrete Introduction*. phiên bản 3. Brooks/-Cole, Cengage Learning, 2013. 506 trang.

[7] Watson S. Stewart J. Clegg D. *Calculus: Early Transcendentals*. phiên bản 9. Cengage Learning, 2011. 1421 trang.

[8] SymPy Development Team. *SymPy Documentation*. phiên bản 1.8. 2750 trang. URL: https://github.com/sympy/sympy/releases.