Học viện Công nghệ Bưu chính Viễn thông
Khoa Công nghệ thông tin 1

Toán rời rạc 1

Giới thiệu

Nguyễn Thị Trang

Vai trò của toán rời rạc trong CNTT

- Là lĩnh vực nghiên cứu cơ bản, đóng vai trò quan trọng trong nhiều lĩnh vực khác của CNTT
  - Trí tuệ nhân tạo
  - Thuật toán
  - Lý thuyết tối ưu
  - ...
- Là môn học cơ sở trong ngành CNTT, nền tảng của nhiều môn học khác
  - Các môn học lập trình
  - Cấu trúc dữ liệu và giải thuật
  - Nhập môn trí tuệ nhân tạo
  - Cơ sở dữ liệu
  - ...

Nội dung môn học (1/5)

- Phần 1: Một số kiến thức cơ bản
  - Lý thuyết tập hợp
  - Logic mệnh đề
  - Thuật toán và độ phức tạp
  - Bài tập

Nội dung môn học (2/5)

- Phần 2: Bài toán đếm
  - Giới thiệu bài toán
  - Các nguyên lý đếm cơ bản
  - Quy về bài toán con
  - Hệ thức truy hồi
  - Phương pháp hàm sinh
  - Bài tập

Nội dung môn học (3/5)

- Phần 3: Bài toán liệt kê
  - Giới thiệu bài toán
  - Phương pháp sinh
  - Phương pháp quay lui
  - Bài tập

Nội dung môn học (4/5)

- Phần 4: Bài toán tối ưu
  - Giới thiệu bài toán
  - Thuật toán duyệt toàn bộ
  - Thuật toán nhánh cận
  - Bài tập

Nội dung môn học (5/5)

- Phần 5: Bài toán tồn tại
  - Giới thiệu bài toán
  - Phương pháp phản chứng
  - Nguyên lý Dirichlet
  - Bài tập

Thông tin môn học

- Giảng viên
  - ThS. Nguyễn Thị Trang, Bộ môn KHMT, Khoa CNTT 1
  - Email: trangnt@ptit.edu.vn, tranguyen.hust117@gmail.com
- Tài liệu tham khảo
  - Nguyễn Duy Phương. Giáo trình Toán rời rạc. Học viện Công nghệ Bưu chính Viễn Thông.
  - Nguyễn Đức Nghĩa, Nguyễn Tô Thành. Toán rời rạc. Nhà xuất bản Giáo dục, 2005.
  - Đỗ Đức Giáo. Toán rời rạc. Nhà xuất bản ĐHQG Hà Nội, 2003.
- Đánh giá
  - Chuyên cần (10%)
  - Bài tập (10%)
  - Kiểm tra giữa kỳ (10%)
  - Thi cuối kỳ (70%)

Thiếu điểm thành phần hoặc nghỉ quá 20% số buổi sẽ không được thi hết môn!!!

Học viện Công nghệ Bưu chính Viễn thông
Khoa Công nghệ thông tin 1

Toán rời rạc 1

Một số kiến thức cơ bản

Nguyễn Thị Trang

Nội dung

- Lý thuyết tập hợp
- Logic mệnh đề
- Thuật toán và độ phức tạp
- Bài tập

Một số ký hiệu tập hợp

- Tập hợp: $A, B, \dots, X, Y, ..$
- Phần tử của tập hợp: $a, b, \dots, x, y, ..$
- Phần tử x thuộc (không thuộc) A: $x \in A, x \notin A$
- Số phần tử của tập hợp
  - Một tập hợp có n phần tử được gọi là một n-tập
- Tập hợp con: $A \subseteq B$
  - $x \in A \Rightarrow x \in B$
- Tập hợp bằng nhau: nếu $A \subseteq B$ và $\mathrm{B} \subseteq A$ thì $A = B$
- Tập rỗng ∅:
  - Không có phần tử nào
  - Là con của mọi tập hợp

Các phép toán trên tập hợp

- Phần bù của A trong X: $\bar{A} = \{x \in X |\ x \notin A\}$
- Hợp của hai tập hợp: $A \cup B = \{x | x \in A\ \ or\ x \in B\}$
- Giao của hai tập hợp: $A \cap B = \{x | x \in A\ and\ x \in B\}$
- Hiệu của hai tập hợp: $A \backslash \mathrm{B} = \{x | x \in A\ and\ x \notin B\}$
- Luật kết hợp:

$$(A \cup B) \cup C = A \cup (B \cup C)$$
$$(A \cap B) \cap C = A \cap (B \cap C)$$

- Luật giao hoán: $A \cup B = B \cup A\ , A \cap B = B \cap A$
- Luật phân bố:

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$
$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

Các phép toán trên tập hợp (2)

- Luật đối ngẫu: $\overline{A \cup B} = \overline{A} \cap \overline{B}, \overline{A \cap B} = \overline{A} \cup \overline{B}$
- Tích Đề các: $\mathrm{A} \times B = \{(a, b) | a \in A, b \in B\}$

Quan hệ

- **Quan hệ:** một quan hệ hai ngôi R trên tập $X, R(X)$ là một tập con các tích Đề các $X \times X$
- Tính chất của quan hệ:
  - Phản xạ: mọi phần tử có quan hệ với chính nó
  - Đối xứng: a có quan hệ với b kéo theo b có quan hệ với a
  - Kéo theo: a có quan hệ với b và b có quan hệ với c kéo theo a có quan hệ c
- Ví dụ:
  - $X = \{1,2,3,4,8\}$
  - $a, b \in X, a$ có quan hệ R đối với b nếu a chia hết cho b
  - $R(\mathrm{X}) = \{(1,1),(2,1),(2,2),(3,1),(3,3),(4,1),(4,2),(4,4),(8,1),(8,4),(8,2),(8,8)\}$
    - Phản xạ, kéo theo, nhưng không đối xứng.

Quan hệ tương đương và phân hoạch

- **Quan hệ tương đương**: là một quan hệ có đủ ba tính chất, phản xạ, đối xứng, và kéo theo
- **Lớp tương đương**: một quan hệ tương đương trên tập hợp sẽ chia tập hợp thành các lớp tương đương
  - Hai phần tử thuộc cùng một lớp có quan hệ với nhau
  - Hai phần tử khác lớp không có quan hệ với nhau
  - Các lớp tương đương phủ kín tập hợp ban đầu
- **Phân hoạch**: là một họ các lớp tương đương (các tập con khác rỗng) của một tập hợp

Một quan hệ tương đương trên tập hợp sẽ xác định một phân hoạch trên tập hợp, ngược lại một phân hoạch bất Kỳ trên tập hợp sẽ tương ứng với một quan hệ tương đương trên nó.

Ví dụ quan hệ tương đương

- Xét $X = \{1,2,\dots,m\}$, m là số nguyên dương $(m > 2)$
- k là số nguyên dương, $1 < k < m$
- Định nghĩa quan hệ R trên X như sau
- Với $a, b \in X$, $\text{aRb} \Leftrightarrow a \equiv b(mod\ k)$
  - a có quan hệ R với b nếu a và b có cùng số dư khi chia cho k
- R là quan hệ tương đương
  - Phản xạ, đối xứng, kéo theo

- Đặt $A_i = \{a \in X | a \equiv i(mod\ k)\}, i = 0,1,\dots,k-1$
- $A_0, A_1, \dots, A_{k-1}$ tạo thành một phân hoạch của X

Example

- X= {1,2,3,4,5,6} Quan hệ tương đương (phản xạ, kéo theo, đối xứng)
- R: N nếu hai số a và b cùng số dư khi chia cho 3
- R(X)_0 ={(3,6), (6,6), (6,3), (3,3)} (dư = 0)
- R(X)_1 = {(1,4), (1,1), (4,1), (4,4)} (dư =1)
- R(X)_2 = {(2,2), (5,2), (2,5), (5,5)} (dư = 2)

- Định nghĩa phân hoạch: k = 3, i (0 -> 2)
- Đặt A_1 = { a thuộc X| a và 1 (3)}) = {4, 1}
- Đặt A_2 = { a thuộc X | a và 2 (3)) = {2, 5}

Nguyên lý cộng

- Nếu A và B là hai tập rời nhau thì:

$$|A \cup B| = |A| + |B|$$

- Nếu $\{A_1, A_2, \dots, A_k\}$ là một phân hoạch của tập hợp X thì:

$$|X| = |A_1| + |A_2| + \cdots + |A_k|$$

- Nếu A là một tập con của X :

$$|\overline{A}| = |\mathrm{X}| - |\mathrm{A}|$$

- Ví dụ:
  - Một đoàn VĐV gồm 2 môn bắn súng và bơi. Nam có 10 người. Số VĐV thi bắn súng là 14. Số nữ VĐV thi bơi bằng số nam VĐV thi bắn súng. Hỏi toàn đoàn có bao nhiêu VĐV.

Nguyên lý cộng

- Ví dụ:
- Chia đoàn thành 2 lớp: Nam và Nữ
- Nữ lại được chia làm 2: thi bắn súng và thi bơi.
- n_nữ_bơi = n_nam_bắn_súng
- n_bắn_súng = n_nam_bắn_súng + n_nữ_bắn_súng
- ==> n_bắn_súng = n_nữ_bơi + n_nữ_bắn_súng
- n_bắn_súng = 14 ➔ n_nữ =14
- N = 14 + 10 =24

Nguyên lý nhân

- Nếu mỗi thành phần a_i của bộ có thứ tự k thành phần $(a_1, a_2, \dots, a_k)$ có n_i khả năng chọn, thì số bộ được tạo ra sẽ là tích các khả năng $n_1 n_2 \dots n_k$
- Hệ quả:
  - $|A_1 \times A_2 \times \cdots \times A_k| = |A_1||A_2| \dots |A_k|$
  - $\left|A^k\right| = |A|^k$
- Ví dụ
  - Có bao nhiêu số nguyên dương có 5 chữ số được thành lập bởi các chữ số 0,1,2?

Nguyên lý nhân

- Vd:
- Chữ số đầu tiên tạo lên bởi 1 và 2
- Các chữ số còn lại: 0, 1, 2
- => 2.3.3.3.3 = 2.3^4

Chỉnh hợp lặp

- **Định nghĩa**: Một chỉnh hợp lặp chập k của n phần tử là một bộ có thứ tự gồm k thành phần, lấy từ n thành phần đã cho. Các thành phần có thể được lặp lại.
- Theo nguyên lý nhân, số chỉnh hợp lặp chập k của n phần tử là n^k
- Ví dụ
  - Tính số tập con của một n-tập

Chỉnh hợp lặp

- VD:
- Giả sử n-tập đã cho là $X = \{x_1, x_2, \ldots, x_n\}$. Biểu diễn mỗi tập con A của tập đã cho X bằng 1 dãy nhị phân độ dài n:
- X = (x_1, x_2, x_3, x_4) => b = (1,0,1,0) > A = {x1, x_3}
- 2*2*2*2 = 2^4 (n=4)
- $b = (b_1, b_2, .. b_n)$
- Trong đó b_i = 1 nếu phần tử x_i thuộc A và b_i = 0 (ngược lại)
- $i = (1,2 \ldots., n)$
- Dãy tập con là 2^n

VD2

- Tập X = { 1, 2, 3, 4}
- Số cách lấy ra k=2 phần tử, có thể lặp lại. N =4
- A = { (1,1), (1,2), (1,3), (1,4),…}
- Phần tử 1 có: 4 cách chọn
- Phần tử 2 có : 4 cách chọn
- ➔ Số cách chọn : 4*4 = n^k

Chỉnh hợp không lặp

- **Định nghĩa**: Một chỉnh hợp không lặp chập k của n phần tử là một bộ có thứ tự gồm k thành phần, lấy từ n thành phần đã cho. Các thành phần không được lặp lại.
- Theo nguyên lý nhân, số chỉnh hợp không lặp chập k của n phần tử là $n(n-1)\ldots(n-k+1)$

- Ví dụ
  - Tính số đơn ánh từ một k-tập vào một n-tập

Chỉnh hợp không lặp

- Vd: *Biểu diễn mỗi đơn ánh bằng bộ ảnh k thành phần, trong đó thành phần thứ i là ảnh của phần tử thứ i (1<= i <= k). Các ảnh không được lặp lại , ta nhận được số cần tìm là n(n-1)…(n-k+1)*

- Vd2: Từ n=5 chữ số (0,1,2,3,5) tính số số nguyên dương có k=4 chữ số, (các chữ số không trùng lặp nhau)
- N= 4.4.3.2

Hoán vị

- **Định nghĩa**: Ta gọi một hoán vị của n phần tử là một cách xếp thứ tự các phần tử đó
- Một hoán vị của n phần tử là một trường hợp riêng của *chỉnh hợp không lặp khi $k = n$*
  - Số hoán vị của n phần tử là $n.(n-1)\dots 2.1 = n!$

- Ví dụ
  - Cần bố trí việc thực hiện n chương trình trên một máy vi tính. Biết rằng các chương trình thực hiện độc lập (không phụ thuộc thứ tự). Hỏi có bao nhiêu cách?

Hoán vị

- Vd:
- Đánh số các chương trình bởi 1,2,…,n Mỗi cách bố trí việc thực hiện các chương trình trên máy có thể biểu diễn một hoán vị của 1,2,..,n . Từ đó suy ra cách bố trí cần tìm là n!

Tổ hợp

- Định nghĩa: Một tổ hợp chập k của n phần tử là một bộ không kể thứ tự gồm k thành phần khác nhau lấy từ n phần tử đã cho.

$$C_n^k = \frac{n!}{k!\,(n-k)!}$$

- Một số tính chất
  - $C_n^k = C_n^{n-k}$
  - $C_n^0 = C_n^n = 1$
  - $C_n^k = C_{n-1}^{k-1} + C_{n-1}^k$
- Nhị thức Newton

$$(x+y)^n = C_n^0 x^n + C_n^1 x^{n-1} y + \cdots + C_n^n = \sum_{k=0}^{n} C_n^k x^{n-k} y^k$$

- X = {1, 2, 3, 4,5}
- Tổ hợp 3 phần tử từ n=5 là : (1,2,3) , (2,3,1) ➔ 1 tổ hợp .

Bài tập 1

- Sử dụng định nghĩa chứng minh một số phép toán trên tập hợp

Bài tập 2

- Cho biết trong các hệ thức dưới đây hệ thức nào là đúng, hệ thức nào là sai?

1) $A \subseteq A \cap B$
2) $C \subseteq (A \cap B) \cup C$
3) $A \cup B \subseteq A \cap B$
4) $A \cap (B \cup A) = A \cap B$
5) $(A \cup B) \backslash (A \cap B) = A \backslash \mathrm{B}$

Bài tập 3

- Ký hiệu Z là tập hợp các số nguyên. Xét hai tập con của Z:

$$A = \{x \in Z: x = 4p - 1 \; với \; một \; p \in Z \; nào \; đó\}$$
$$B = \{y \in Z: y = 4q + 3 \; với \; một \; q \in Z \; nào \; đó\}$$

Chứng minh rằng $A = B$.

Bài tập 4

- Cho $A = \{0, 1, 2, 3, 4\}$ và xác định quan hệ R trên A bởi:

  $R = \{(0,0), (2,1), (0,3), (1,1), (3,0), (1,4), (4,1), (2,2), (2,4), (3,3), (4,4), (1,2), (4,2)\}.$

1) R có phải là một quan hệ tương đương hay không?
2) Nếu câu 1) là đúng thì chỉ ra phân hoạch của A thành các lớp tương đương theo quan hệ R.

Nội dung

- Lý thuyết tập hợp
- Logic mệnh đề
- Logic vị từ
- Thuật toán và độ phức tạp
- Bài tập

Một số khái niệm của logic mệnh đề

- **Mệnh đề**: là một câu khẳng định hoặc đúng hoặc sai.
- Ví dụ:
  - "Hà Nội là thủ đô của Việt Nam" là một mệnh đề đúng.
  - "(5 < 3)" là một mệnh đề sai, "(5 > 3)" là một mệnh đề đúng.
  - "(a<7)" không phải là mệnh đề vì nó không biết khi nào đúng khi nào sai.
- **Giá trị chân lý của mệnh đề**: mỗi mệnh đề chỉ có một trong 2 giá trị "**đúng**", ký hiệu là "**T**", giá trị "**sai**", ký hiệu là "**F**". Tập { T, F} được gọi là giá trị chân lý của mệnh đề.
- Ký hiệu**:** ta ký hiệu mệnh đề bằng các chữ cái in thường (a, b, p, q, r, s, t). *Mỗi* mệnh đề còn được gọi là một công thức. Từ khái niệm về mệnh đề, giá trị chân lý của mỗi mệnh đề, ta xây dựng nên các mệnh đề phức hợp (được gọi là công thức) thông qua các phép toán trên mệnh đề.

Các phép toán của logic mệnh đề (1/2)

Cho p và q là hai mệnh đề

- Phép phủ định
  - $\neg p$ là ký hiệu mệnh đề, đọc là "Không phải p"
  - Mệnh đề cho giá trị đúng nếu p sai và cho giá trị sai nếu p đúng
- Phép hội
  - $p \wedge q$ là ký hiệu mệnh đề, đọc là "p và q"
  - Mệnh đề có giá trị đúng khi cả p và q có giá trị đúng, có giá trị sai trong các trường hợp khác còn lại
- Phép tuyển
  - $p \vee q$ là ký hiệu mệnh đề, đọc là "p hoặc q"
  - Mệnh đề có giá trị sai khi cả p và q có giá trị sai, có giá trị đúng trong các trường hợp khác còn lại

Các phép toán của logic mệnh đề (2/2)

- Phép tuyển loại
  - $p \oplus q$ là ký hiệu mệnh đề đọc là "hoặc p hoặc q"
  - Mệnh đề có giá trị đúng khi một trong p hoặc q có giá trị đúng, có giá trị sai trong các trường hợp khác còn lại
- Phép kéo theo
  - $p \Rightarrow q$ là ký hiệu mệnh đề đọc là "p kéo theo q"
  - Mệnh đề có giá trị sai khi p đúng và q sai, có giá trị đúng trong các trường hợp khác còn lại
- Phép tương đương
  - $p \Leftrightarrow q$ là ký hiệu mệnh đề đọc là "p tương đương q"
  - Mệnh đề có giá trị đúng khi p và q có cùng giá trị chân lý, có giá trị sai trong các trường hợp khác còn lại

Bảng giá trị chân lý

Bảng giá trị chân lý các phép toán mệnh đề							
p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \oplus q$	$p \Rightarrow q$	$p \Leftrightarrow q$
T	T	F	T	T	F	T	T
T	F	F	F	T	T	F	F
F	T	T	F	T	T	T	F
F	F	T	F	F	F	T	T

Các phép toán cấp bít ứng dụng trong ngôn ngữ LT				
Giá trị của A	Giá trị của B	A **and** B	A **or** B	A **xor** B
A = 13 =1101	B = 8=1000	1000	1101	0101

Một số khái niệm

- Thỏa được
  - Một mệnh đề là thỏa được nếu nó đúng với một bộ giá trị chân lý nào đó của các mệnh đề thành phần
- Không thỏa được
  - Một mệnh đề là không thỏa được nếu nó sai với mọi bộ giá trị chân lý của các mệnh đề thành phần
- Vững chắc
  - Một mệnh đề là vững chắc nếu nó đúng với mọi bộ giá trị chân lý của các mệnh đề thành phần

Các mệnh đề tương đương logic (1/2)

- Hai mệnh đề a và b được gọi là tương đương logic nếu chúng có cùng giá trị chân lý với mọi bộ giá trị chân lý của các mệnh đề thành phần
  - Ký hiệu: $a \equiv b$

- Một số mệnh đề tương đương cơ bản
  - $a \vee F \equiv a$
  - $a \wedge F \equiv F$
  - $a \vee T \equiv T$
  - $a \wedge T \equiv a$
  - $a \Rightarrow b \equiv \neg a \vee b$
  - $a \Leftrightarrow b \equiv (a \Rightarrow b) \wedge (b \Rightarrow a)$
  - $\neg(\neg a) \equiv a$

Các mệnh đề tương đương logic (2/2)

- Luật giao hoán
  - $a \vee b \equiv b \vee a$
  - $a \wedge b \equiv b \wedge a$
- Luật kết hợp
  - $(a \vee b) \vee \mathrm{c} \equiv a \vee (b \vee \mathrm{c})$
  - $(a \wedge b) \wedge \mathrm{c} \equiv a \wedge (b \wedge \mathrm{c})$
- Luật phân phối
  - $a \wedge (b \vee \mathrm{c}) \equiv (a \wedge b) \vee (a \wedge c)$
  - $a \vee (b \wedge \mathrm{c}) \equiv (a \vee b) \wedge (a \vee c)$
- Luật De Morgan
  - $\neg(a \vee b) \equiv \neg a \wedge \neg b$
  - $\neg(a \wedge b) \equiv \neg a \vee \neg b$

Dạng chuẩn tắc hội (1/2)

- Một câu (mệnh đề) tuyển là tuyển của các mệnh đề nguyên thủy
  - Câu tuyển có dạng $p_1 \vee p_2 \vee ... \vee p_n$ trong đó p_i là các mệnh đề nguyên thủy

- Một công thức ở dạng chuẩn tắc hội nếu nó là hội của các câu tuyển
  - $(a \vee e \vee f \vee g) \wedge (b \vee c \vee d)$

Dạng chuẩn tắc hội (2/2)

- Ta có thể biến đổi một công thức bất kỳ về dạng chuẩn tắc hội bằng cách biến đổi theo nguyên tắc sau:
  - Khử các phép tương đương: $a \Leftrightarrow b \equiv (a \Rightarrow b) \land (b \Rightarrow a)$
  - Khử các phép kéo theo: $a \Rightarrow b \equiv \neg a \lor b$
  - Chuyển các phép phủ định vào sát các ký hiệu mệnh đề bằng cách áp dụng luật De Morgan
  - Khử phủ định kép: $\neg(\neg a) \equiv a$
  - Áp dụng luật phân phối: $a \lor (b \land \mathrm{c}) \equiv (a \lor b) \land (a \lor c)$

Bài tập 1

- Sử dụng phương pháp bảng chân lý chứng minh sự tương đương logic giữa các mệnh đề
  - Các mệnh đề tương đương cơ bản
  - Các luật

- Sử dụng các mệnh đề tương đương cơ bản và các luật (giao hoán, kết hợp, phân phối, De Morgan) chứng minh sự tương đương logic giữa các mệnh đề

$$\neg(p \vee (\neg p \wedge q)) \equiv \neg p \wedge \neg q$$

Bài tập 2

- Chứng minh các mệnh đề sau là vững chắc

$a)\quad (p \wedge q) \Rightarrow p$

$b)\quad p \Rightarrow (p \vee q)$

$c)\quad \neg p \Rightarrow (p \Rightarrow q)$

$d)\quad (p \wedge q) \Rightarrow (p \Rightarrow q)$

$e)\quad \neg(p \Rightarrow q) \Rightarrow p$

$f)\quad \neg(p \Rightarrow q) \Rightarrow \neg q$

g) $\neg p \wedge (p \vee q) \Rightarrow q$

h) $(p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$

i) $(p \wedge (p \Rightarrow q)) \Rightarrow q$

j) $((p \vee q) \wedge (p \Rightarrow r) \wedge (q \Rightarrow r)) \Rightarrow r$

Bài tập 3

- Chứng minh các tương đương logic sau

1) $(p \Leftrightarrow q) \equiv (p \land q) \lor (\neg p \land \neg q)$

2) $\neg p \Leftrightarrow q \equiv p \Leftrightarrow \neg q$

3) $\neg(p \Leftrightarrow q) \equiv \neg p \Leftrightarrow q$

Bài tập 4

- Chuẩn hóa về dạng chuẩn tắc hội

$$(p \Rightarrow q) \vee \neg(r \vee \neg s)$$

Nội dung

- Lý thuyết tập hợp
- Logic mệnh đề
- Thuật toán và độ phức tạp
- Bài tập

Khái niệm thuật toán

- Thuật toán hoặc giải thuật (algorithm)
  - Là một thủ tục giải quyết một vấn đề nào đó trong một số hữu hạn bước
  - Là một tập hữu hạn các chỉ thị được định nghĩa rõ ràng để giải quyết một vấn đề nào đó
  - Là một tập các quy tắc định nghĩa chính xác một dãy các hành động
- Mô tả thuật toán
  - Sử dụng ngôn ngữ tự nhiên
  - Sử dụng dạng giả mã (Pseudo-code)
  - Sử dụng ngôn ngữ lập trình

Ví dụ thuật toán (1/2)

- **Thuật toán**: Tìm số nguyên lớn nhất trong danh sách n số nguyên (chưa sắp xếp)

- Mô tả thuật toán sử dụng ngôn ngữ tự nhiên
  - Nếu không có số nào trong danh sách thì không có số lớn nhất
  - Giải sử số đầu tiên là số lớn nhất trong danh sách
  - Với mỗi số còn lại trong danh sách, nếu số này lớn hơn số lớn nhất hiện tại thì coi số này là số lớn nhất trong danh sách
  - Khi tất cả các số trong danh sách đều đã được xem xét, số lớn nhất hiện tại là số lớn nhất trong danh sách

Dài dòng, ít được sử dụng

Ví dụ thuật toán (2/2)

- **Thuật toán**: Tìm số nguyên lớn nhất trong danh sách n
- số nguyên (chưa sắp xếp)

- Mô tả thuật toán sử dụng dạng giả mã

- **Algorithm** LargestNumber
- **Input:** A list of numbers L.
- **Output:** The largest number in the list L.

  - *largest* ← null
    - **for each** *item* **in** L **do**
    if *item* > *largest* **then**
      - *largest* ← *item*
  return *largest*

Dễ hiểu, hay được sử dụng,
không phụ thuộc vào ngôn ngữ lập trình

Độ phức tạp thuật toán

- Hầu hết các thuật toán được thiết kế làm việc với kích thước dữ liệu đầu vào tùy ý
  - Ví dụ thuật toán ở trang trước, kích thước dữ liệu đầu vào là số phần tử trong danh sách (n)

- Độ phức tạp thời gian (time complexity)
  - Xác định lượng thời gian cần thiết để thực hiện giải thuật
    - Được tính là số phép toán cơ bản thực hiện giải thuật
- Độ phức tạp không gian (space complexity)
  - Xác định lượng bộ nhớ cần thiết để thực hiện giải thuật
    - Lượng bộ nhớ lớn nhất cần thiết để lưu các đối tượng của thuật toán tại một thời điểm thực hiện thuật toán

Thường được biểu diễn như là một hàm của kích thước dữ liệu đầu vào

Khái niệm O-lớn

- $f(n) = O(g(n))$, với n đủ lớn, $f(n)$ không quá một hằng số cố định nhân với $g(n)$
  - n là một số nguyên dương kí hiệu kích thước của dữ liệu đầu vào

$$f(n) = O\big(g(n)\big) \Leftrightarrow \exists c, n_0. \forall n \geq n_0, f(n) \leq c * g(n).$$

- Định lý
  - Nếu $lim_{n\to\infty} \frac{f(n)}{g(n)}$ tồn tại và hữu hạn, thì $f(n) = O\big(g(n)\big)$

Ví dụ về độ phức tạp thuật toán

- **Thuật toán**: Tìm số nguyên lớn nhất trong danh sách n số nguyên

Algorithm LargestNumber
Input: A list of numbers L.
Output: The largest number in the list L.

largest ← null
for each *item* **in** L **do**
    if *item* > *largest* **then**
        largest ← *item*
return *largest*

Độ phức tạp thời gian và không gian đều là $O(n)$

Bài tập 1

- Tìm độ phức tạp tính toán của thuật toán sắp xếp nổi bọt (bubble sort)

```
function bubble_sort(List L, number n) // n chiều dài L
    for i from n downto 2
        for j from 1 to (i - 1)
            if L[j] > L[j + 1] then //nếu chúng không đúng thứ tự
                swap(L[j], L[j + 1]) //đổi chỗ chúng cho nhau
            end if
        end for
    end for
end function
```

Bài tập 2

- Tìm độ phức tạp tính toán của thuật toán tìm kiếm nhị phân trên danh sách đã sắp xếp

function binary_search(A, x, L, R)
    if $L > R$ **then**
        return Fail
    else
        $i \leftarrow \lfloor (L + R)/2 \rfloor$
        if $A[i] == x$ **then**
            return i
        else if $A[i] > x$ **then**
            return binary_search($A, x, L, i - 1$)
        else
            return binary_search($A, x, i + 1, R$)
        end if
    end if
end function

Học viện Công nghệ Bưu chính Viễn thông
Khoa Công nghệ thông tin 1

Toán rời rạc 1

Bài toán đếm

Nguyễn Thị Trang

Nội dung

- Giới thiệu bài toán
- Các nguyên lý cơ bản
- Quy về bài toán con
- Hệ thức truy hồi
- Phương pháp hàm sinh
- Bài tập

Giới thiệu bài toán đếm

- Bài toán đếm
  - Là bài toán đếm xem có bao nhiêu cấu hình tổ hợp có thể được tạo ra với những quy tắc đã nêu?
  - Lời giải thường phụ thuộc vào một số tham số ban đầu và người ta cố gắng biểu diễn những phụ thuộc này bằng những công thức toán học.
- Nguyên tắc chung giải bài toán đếm
  - Để đếm cấu hình đã cho, người ta tìm cách đưa về các cấu hình quen thuộc bằng cách thiết lập một quan hệ 1-1 giữa chúng
- Ứng dụng của bài toán đếm trọng khoa học máy tính.
  - Ước lượng số phép toán thực hiện trong một giải thuật, chương trình máy tính.
  - Ước lượng độ phức tạp thời gian và không gian của giải thuật.

Các phương pháp giải quyết bài toán đếm

- Sử dụng nguyên lý đếm cơ bản: Nguyên lý cộng, nguyên lý nhân, nguyên lý bù trừ
- Quy về các bài toán con: Phân tích lời giải bài toán đếm phức tạp thành những bài toán con. Trong đó, mỗi bài toán con có thể giải được bằng các nguyên lý đếm cơ bản.
- Sử dụng hệ thức truy hồi: Xây dựng công thức tính số nghiệm tổng quát bất kì dựa vào biểu diễn các số hạng biết trước.
- Phương pháp hàm sinh: Sử dụng hàm sinh của một dãy số để đếm cấu hình tổ hợp.

Nội dung

- Giới thiệu bài toán
- Các nguyên lý cơ bản
- Quy về bài toán con
- Hệ thức truy hồi
- Phương pháp hàm sinh
- Bài tập

Nguyên lý cộng (nhắc lại)

- Nếu A và B là hai tập rời nhau thì
$$|A \cup B| = |A| + |B|$$
- Nếu $\{A_1, A_2, \ldots, A_k\}$ là một phân hoạch của tập hợp X thì
$$|X| = |A_1| + |A_2| + \cdots + |A_k|$$
- Nếu có K việc, việc thứ i thực hiện bằng n_i cách và thực hiện một cách tuần tự. Khi đó sẽ có $n_1 + n_2 +..+ n_K$ cách thực hiện một trong K việc nêu trên.

Ví dụ 1

- **Bài toán**: Giả sử N, M là hai số tự nhiên đã xác định giá trị. Hãy cho biết giá trị của S sau khi thực hiện đoạn chương trình.

$$
\begin{aligned}
&S = 0;\\
&\textbf{for } (i = 1;\ i <= N; i++)\\
&\qquad S++;\\
&\textbf{for } (j = 1;\ j <= M; j++)\\
&\qquad S++;
\end{aligned}
$$

Ví dụ 1

- **Bài toán**: Giả sử N, M là hai số tự nhiên đã xác định giá trị. Hãy cho biết giá trị của S sau khi thực hiện đoạn chương trình.

  $S = 0;$
  for $(i = 1;\ i <= N; i++)$
  $\quad S++;$
  for $(j = 1;\ j <= M; j++)$
  $\quad S++;$

- **Lời giải**: Gọi số phép toán thực hiện trong vòng lặp thứ nhất là T_1, số phép toán thực hiện trong vòng lặp thứ hai là T_2. Vì hai vòng lặp thực hiện độc lập nhau nên theo nguyên lý cộng, giá trị của $S = T_1 + T_2 = N + M$.

Nguyên lý nhân (nhắc lại)

- Giả sử một nhiệm vụ nào đó được tách ra làm hai việc. Việc thứ nhất có thể thực hiện bằng n_1 cách, việc thứ hai thực hiện bằng n_2 cách sau khi việc thứ nhất đã được thực hiện. Khi đó, sẽ có $n_1 n_2$ cách thực hiện nhiệm vụ nêu trên.
- Nếu mỗi thành phần a_i của bộ có thứ tự k thành phần $(a_1, a_2, \dots, a_k)$ có n_i khả năng chọn, thì số bộ được tạo ra sẽ là tích các khả năng $n_1 n_2 \dots n_k$
- Hệ quả:
  - $|A_1 \times A_2 \times \cdots \times A_k| = |A_1||A_2| \dots |A_k|$
  - $|A^k| = |A|^k$

Ví dụ 2

- **Bài toán**: Giả sử n_1, n_2 là hai số nguyên dương đã xác định giá trị. Hãy cho biết giá trị của S sau khi thực hiện đoạn chương trình dưới đây?

```
int S = 0;
for (int i = 1; i <= n1; i++)
    for (int j = 1; j <= n2; j++)
        S++;
```

- **Lời giải.** Với mỗi giá trị của $i = 1,2,..,n_1$ thì S được cộng n_2 đơn vị. Do vậy, theo nguyên lý nhân, sau n_1 vòng lặp giá trị của $S = n_1 \times n_2$.

Ví dụ 3

- **Bài toán**: Có bao nhiêu số nguyên dương có 5 chữ số không chứa chữ số 1 và không có 2 chữ số nào giống nhau?

Ví dụ 3

- **Bài toán**: Có bao nhiêu số nguyên dương có 5 chữ số không chứa chữ số 1 và không có 2 chữ số nào giống nhau?

- **Lời giải**:
  - Xét số có 5 chữ số $a_1a_2a_3a_4a_5$
  - a_1 có 8 cách chọn (trừ 0 và 1)
  - a_2 có 8 cách chọn (trừ a_1 và 1)
  - a_3 có 7 cách chọn (trừ a_1, a_2, và 1)
  - a_4 có 6 cách chọn (trừ a_1, a_2, a_3, và 1)
  - a_5 có 5 cách chọn (trừ a_1, a_2, a_3, a_4, và 1)
  - Vậy có 8x8x7x6x5 số thỏa mãn

Ví dụ 4

- **Bài toán**: có bao nhiêu tên biến trong ngôn ngữ lập trình C độ dài 8 chỉ chứa hai chữ cái a,b và bắt đầu bởi aaa hoặc bbb?

Ví dụ 4

- **Bài toán**: có bao nhiêu tên biến trong ngôn ngữ lập trình C độ dài 8 chỉ chứa hai chữ cái a,b và bắt đầu bởi aaa hoặc bbb?
- **Lời giải**: Tập các biến thỏa mãn đề bài được phân hoạch làm 2 tập: một tập gồm các biến bắt đầu bằng aaa, tập kia gồm các biến bắt đầu bằng bbb. Mỗi tên biến độ dài 8 bắt đầu bằng aaa (hoặc bbb) có thể được xây dựng như sau:
  - Chọn ký tự thứ 4, chọn ký tự thứ 5, …, chọn ký tự thứ 8.
  - Mỗi ký tự có 2 cách chọn: a hoặc b
  - Có tất cả $2 \times 2 \times 2 \times 2 \times 2 = 32$ cách
- Toàn bộ có: $32 + 32 = 64$ biến

Hoán vị, chỉnh hợp, tổ hợp (nhắc lại)

- Số hoán vị của n phần tử: $n.(n-1)\dots 2.1 = n!$
- Số chỉnh hợp lặp chập k của n phần tử là n^k
- số chỉnh hợp không lặp chập k của n phần tử là $n(n-1)\dots(n-k+1) = \frac{n!}{(n-k)!}$
- Số tổ hợp chập k của n phần tử là

$$C_n^k = \frac{n!}{k!\,(n-k)!}$$

Ví dụ 5

- **Bài toán**: Phương trình $x_1 + x_2 + x_3 = 11$ có bao nhiêu nghiệm nguyên không âm?

Ví dụ 5

- **Bài toán**: Phương trình $x_1 + x_2 + x_3 = 11$ có bao nhiêu nghiệm nguyên không âm?

- **Lời giải**: Số nghiệm nguyên không âm của phương trình bằng số cách chọn 11 phần tử từ 3 tập khác nhau. Ta biểu diễn 11 phần tử bằng 11 số 1 trên một đường thẳng. Sau đó sử dụng 2 số 0 để chia 11 phần tử này ra thành 3 nhóm. Số số 1 trong mỗi nhóm tương ứng số phần tử ta sẽ chọn từ tập tương ứng. Như vậy số nghiệm của phương trình chính là số cách chọn 11 vị trí để đánh số 1 trong một dãy 13 vị trí (để đánh 0 và 1).

$$C_{13}^{11} = \frac{13 \times 12}{2} = 78$$

Ví dụ 6

- **Bài toán**: Phương trình $x_1 + x_2 + \cdots + x_n = k$ có bao nhiêu nghiệm nguyên không âm?

Ví dụ 6

- **Bài toán**: Phương trình $x_1 + x_2 + \cdots + x_n = k$ có bao nhiêu nghiệm nguyên không âm?

- **Lời giải:** Tương tự Ví dụ 5 ta sẽ có số nghiệm nguyên không âm của phương trình là:

$$C_{n-1+k}^{k} = \frac{(n-1+k)!}{k!\,(n-1)!}$$

Ví dụ 7

- **Bài toán:** Phương trình $x_1 + x_2 + x_3 = 11$ có bao nhiêu nghiệm nguyên không âm thỏa mãn $x_1 \geq 1, x_2 \geq 2, x_3 \geq 3$.

Ví dụ 7

- **Bài toán:** Phương trình $x_1 + x_2 + x_3 = 11$ có bao nhiêu nghiệm nguyên không âm thỏa mãn $x_1 \geq 1, x_2 \geq 2, x_3 \geq 3$.

- **Lời giải**: Phương trình tương đương:

$$(x_1 - 1) + (x_2 - 2) + (x_3 - 3) = 5$$

Đặt $y_1 = x_1 - 1, y_2 = x_2 - 2$, $y_3 = x_3 - 3$

Phương trình trở thành: $y_1 + y_2 + y_3 = 5$

Theo Ví dụ 6, số nghiệm nguyên không âm là

$$C_{3-1+5}^{5} = C_7^5 = \frac{7 \times 6}{2} = 21$$

Ví dụ 8

- **Bài toán**: Phương trình $x_1 + x_2 + \cdots + x_n = k$ có bao nhiêu nghiệm nguyên không âm thỏa mãn $x_1 \geq m_1, \ldots, x_n \geq m_n$?

Ví dụ 8

- **Bài toán**: Phương trình $x_1 + x_2 + \cdots + x_n = k$ có bao nhiêu nghiệm nguyên không âm thỏa mãn $x_1 \geq m_1, \ldots, x_n \geq m_n$?

- **Lời giải**: Phương trình tương đương

$(x_1 - m_1) + \cdots + (x_n - m_n) = k - (m_1 + \cdots + m_n)$ Đặt $m = k - (m_1 + \cdots + m_n)$

Bài toán quy về tìm số nghiệm nguyên không âm của phương trình: $y_1 + y_2 + \cdots + y_n = \mathrm{m}$

Theo ví dụ 6: C_{n-1+m}^{m}

Ví dụ 9

Bài toán: Phương trình $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 24$ có bao nhiêu nghiệm nguyên không âm thỏa mãn $1 \leq x_1 \leq 5, 3 \leq x_2 \leq 7$?

Ví dụ 9

Bài toán: Phương trình $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 24$ có bao nhiêu nghiệm nguyên không âm thỏa mãn $1 \le x_1 \le 5$, $3 \le x_2 \le 7$?

Lời giải: Gọi N_1, N_2, N_3, N_4 là số các nghiệm nguyên không âm của phương trình (1), (2), (3), (4).

$$\begin{cases} x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 24 \\ x_1 \ge 1, x_2 \ge 3 \end{cases} \qquad (1)$$

$$\begin{cases} x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 24 \\ x_1 \ge 6, x_2 \ge 3; \end{cases} \qquad (2)$$

$$\begin{cases} x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 24 \\ x_1 \ge 1, x_2 \ge 8; \end{cases} \qquad (3)$$

$$\begin{cases} x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 24 \\ x_1 \ge 6, x_2 \ge 8; \end{cases} \qquad (4)$$

Ví dụ 9

TheoVí dụ 8 ta có:

$$N_1 = C_{6-1+20}^{20} = C_{25}^{20} = 53130$$
$$N_2 = C_{6-1+15}^{15} = C_{20}^{15} = 15504$$
$$N_3 = C_{6-1+15}^{15} = C_{20}^{15} = 15504$$
$$N_4 = C_{6-1+10}^{10} = C_{15}^{10} = 3003$$

Vì vậy số nghiệm thỏa mãn yêu cầu bài toán là:

$$\begin{aligned} N &= N_1 - N_2 - N_3 + N_4 \\ &= 53130 - 15504 - 15504 + 3003 \\ &= 25125 \end{aligned}$$

Bài tập 1

- Một đợt phát hành xổ số mỗi tấm vé số gồm 2 phần. Phần chữ gồm 2 chữ cái nhận giá trị từ A đến Z, phần số gồm 4 chữ số nhận giá trị từ 0 đến 9. Mỗi đợt phát hành như vậy có 1 giải đặc biệt, 2 giải nhất, 5 giải nhì, và 10 giải ba. Tính xác suất 1 tấm vé số trúng giải từ giải ba trở lên trong 2 trường hợp sau

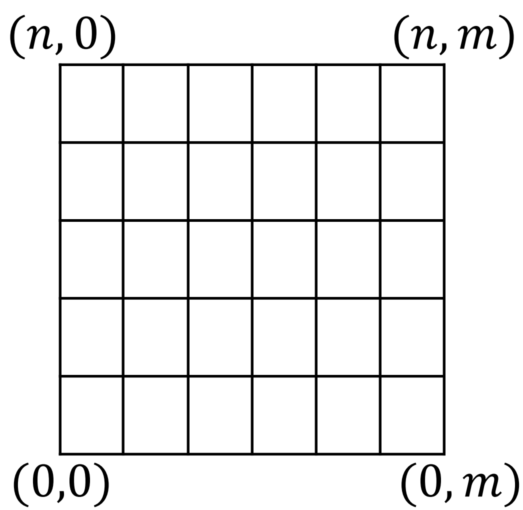
a. Phần chữ đứng trước phần số trong mỗi tấm vé số.
b. Phần chữ đứng tùy ý trong mỗi tấm vé số.

Bài tập 2

- Tính số trận đấu tại mỗi vòng chung kết World Cup bóng đá.

Bài tập 3

- Cho lưới như hình vẽ. Ta đánh số các cột từ 0 tới m (theo chiều từ trái sang phải) và các hàng từ 0 tới n (theo chiều từ dưới lên trên). Hỏi có bao nhiêu cách di chuyển từ vị trí (0,0) tới vị trí (n,m) nếu ta chỉ di chuyển dọc theo cạch các ô theo chiều từ trái sang phải và theo chiều từ dưới lên trên.



Nguyên lý bù trừ

- Nhiều bài toán đếm phức tạp hơn có thể giải bằng nguyên lý bù trừ
- Về bản chất, nguyên lý bù trừ là trường hợp tổng quát của nguyên lý cộng
- **Nguyên lý bù trừ**: Nếu A và B là hai tập hợp, khi đó:

$$|A \cup B| = |A| + |B| - |A \cap B|$$

- Tổng quát, nếu $A_1, A_2, \ldots, A_k$ là các tập hợp hữu hạn, khi đó:

$$|A_1 \cup A_2 \cup \ldots \cup A_k| = N_1 - N_2 + \ldots + (-1)^{k-1} N_k,$$

Trong đó, N_i là tổng của tất cả các giao của i tập lấy từ k tập đã cho

Ví dụ 10

- **Bài toán**: Trong tập hợp $X = \{1,2, \ldots , 10000\}$ có bao nhiêu số không chia hết cho bất kỳ số nào trong các số 3, 4, 7?

Ví dụ 10

- **Bài toán**: Trong tập hợp $X = \{1,2, \dots , 10000\}$ có bao nhiêu số không chia hết cho bất kỳ số nào trong các số 3, 4, 7?

  Lời giải:

  Gọi A_1 là tập các số thuộc X và chia hết cho 3

  A_2 là tập các số thuộc X và chia hết cho 4

  A_3 là tập các số thuộc X và chia hết cho 7

  Khi đó

  $$|A_1 \cup A_2 \cup A_3| = |A_1| + |A_2| + |A_3| - |A_1 \cap A_2| - |A_1 \cap A_3| - |A_2 \cap A_3| + |A_1 \cap A_2 \cap A_3|$$

Ví dụ 10

Tính toán trực tiếp các giá trị ta có:

$N_1 = |A_1| + |A_2| + |A_3|$

$= [10000/3] + [10000/4] + [10000/7]$

$= 3333 + 2500 + 1428 = 7261.$

$N_2 = |A_1 \cap A_2| + |A_1 \cap A_3| + |A_2 \cap A_3|$

$= [10000/(3 * 4)] + [10000/(3 * 7)] + [10000/(4 * 7)]$

$= 833 + 476 + 357 = 1666.$

$N_3 = |A_1 \cap A_2 \cap A_3| = [10000/(3 * 4 * 7)] = 119$

Từ đó ta có số các số hoặc chia hết cho 3, hoặc chia hết cho 4, hoặc chia hết cho 7 là

$$N = N_1 - N_2 + N_3 = 7261 - 1666 + 119 = 5714.$$

Như vậy, số các số không chia hết cho bất kỳ số 3, 4, 7 là

$$K = 10000 - N = 4286.$$

Ví dụ 11

- **Bài toán**: Biết rằng có 1232 sinh viên học tiếng Tây Ban Nha, 879 sinh viên học tiếng Pháp và 114 sinh viên học tiếng Nga, 103 sinh viên học cả tiếng Tây Ban Nha và tiếng Pháp, 23 sinh viên học tiếng Tây Ban Nha và tiếng Nga, 14 sinh viên học cả tiếng Pháp và tiếng Nga. Nếu tất cả có 2092 sinh viên theo học ít nhất một ngoại ngữ thì có bao nhiêu sinh viên học cả ba thứ tiếng.

Ví dụ 12

- **Bài toán bỏ thư**: Có *n* lá thư bỏ vào *n* phong bì ghi sẵn địa chỉ. Bỏ ngẫu nhiên các lá thư vào các phong bì. Hỏi xác suất để xảy ra không một lá thư nào bỏ đúng địa chỉ là bao nhiêu?

Ví dụ 12

- **Bài toán bỏ thư**: Có *n* lá thư bỏ vào *n* phong bì ghi sẵn địa chỉ. Bỏ ngẫu nhiên các lá thư vào các phong bì. Hỏi xác suất để xảy ra không một lá thư nào bỏ đúng địa chỉ là bao nhiêu?
- **Gợi ý**: Gọi A_i là tập các cách bỏ thư thỏa mãn lá thư thứ i đúng địa chỉ. Số cách bỏ thư thỏa mãn yêu cầu

$$N - |A_1 \cup A_1 \cup \ldots \cup A_n| = N - N_1 + N_2 \ldots + (-1)^n N_n$$

Trong đó $N = n!$ là số cách bỏ thư, N_k là số tất cả các cách bỏ thư sao cho có k lá thư bỏ đúng địa chỉ.

Có C_n^k cách chọn ra k lá thư đúng địa chỉ, với mỗi cách chọn ra k lá thư đúng địa chỉ có $(n-k)!$ cách bỏ các lá thư còn lại. Do vậy $N_k = C_n^k(n-k)! = \frac{n!}{k!}$

- Xác suất: $\frac{N - N_1 + N_2 \ldots + (-1)^n N_n}{N}$

Bài tập 1

- Có bao nhiêu cách xếp 5 người, A, B, C, D, E, đứng thành một hàng ngang sao cho A không đứng cạnh B?

Bài tập 2

- Tính số lượng số có 5 chữ số sao cho có ít nhất hai chữ số giống nhau?

Nội dung

- Giới thiệu bài toán
- Các nguyên lý đếm cơ bản
- Quy về bài toán con
- Hệ thức truy hồi
- Phương pháp hàm sinh
- Bài tập

Quy về bài toán con

- Một phương pháp khác để giải bài toán đếm là quy về các bài toán con đơn giản hơn
  - Điều này không phải lúc nào cũng dễ dàng vì ta cần phải phân tích sâu sắc các cấu hình cần đếm

Ví dụ 13

- **Bài toán**: Trong các số tự nhiên có 7 chữ số hãy đếm số các số thuận nghịch (số đối xứng) có tổng các chữ số là 18?

Ví dụ 13

- **Bài toán**: Trong các số tự nhiên có 7 chữ số hãy đếm số các số thuận nghịch (số đối xứng) có tổng các chữ số là 18?
- **Lời giải**: Một số thỏa mãn đề bài có dạng $x_1x_2x_3x_4x_3x_2x_1 (x_1 \geq 1)$ và

$$2x_1 + 2x_2 + 2x_3 + x_4 = 18$$

- Vì $2x_1, 2x_2, 2x_3$ là những số chẵn nên x_4 cũng phải là một số chẵn. Do đó x_4 có thể nhận các giá trị 0, 2, 4, 6, 8.
- Gọi N_0, N_2, N_4, N_6, N_8 là số nghiệm của pt ứng với các trường hợp x_4 nhận giá trị 0, 2, 4, 6, 8. Theo nguyên lý cộng số cần tìm là

$$N = N_0 + N_2 + N_4 + N_6 + N_8$$

Ví dụ 13

- Ta có N_0, N_2, N_4, N_6, N_8 là số nghiệm của các pt tương ứng sau

$$\begin{cases} 2x_1 + 2x_2 + 2x_3 + 0 = 18 \\ x_1 \geq 1 \end{cases} \Leftrightarrow \begin{cases} x_1 + x_2 + x_3 = 9 \\ x_1 \geq 1 \end{cases}$$

$$\begin{cases} 2x_1 + 2x_2 + 2x_3 + 2 = 18 \\ x_1 \geq 1 \end{cases} \Leftrightarrow \begin{cases} x_1 + x_2 + x_3 = 8 \\ x_1 \geq 1 \end{cases}$$

$$\begin{cases} 2x_1 + 2x_2 + 2x_3 + 4 = 18 \\ x_1 \geq 1 \end{cases} \Leftrightarrow \begin{cases} x_1 + x_2 + x_3 = 7 \\ x_1 \geq 1 \end{cases}$$

$$\begin{cases} 2x_1 + 2x_2 + 2x_3 + 6 = 18 \\ x_1 \geq 1 \end{cases} \Leftrightarrow \begin{cases} x_1 + x_2 + x_3 = 6 \\ x_1 \geq 1 \end{cases}$$

$$\begin{cases} 2x_1 + 2x_2 + 2x_3 + 8 = 18 \\ x_1 \geq 1 \end{cases} \Leftrightarrow \begin{cases} x_1 + x_2 + x_3 = 5 \\ x_1 \geq 1 \end{cases}$$

Các bài toán con này có thể giải dễ dàng theo Ví dụ 8

Nội dung

- Giới thiệu bài toán
- Các nguyên lý đếm cơ bản
- Quy về bài toán con
- Hệ thức truy hồi
- Phương pháp hàm sinh
- Bài tập

Hệ thức truy hồi

- **Định nghĩa**: Hệ thức truy hồi đối với dãy số $\{a_n\}$ là công thức biểu diễn a_n qua một hay nhiều số hạng đi trước của dãy, cụ thể là $a_1, a_2, \ldots, a_{n-1}$, với n nguyên và $n \geq n_0$, trong đó n_0 là nguyên không âm. Dãy số được gọi là lời giải hay nghiệm của hệ thức truy hồi nếu các số hạng của nó thỏa mãn hệ thức truy hồi này.
- **Ví dụ 14**: Cho $\{a_n\}$ là dãy số thỏa mãn hệ thức truy hồi $a_n = a_{n-1} - a_{n-2}$, với $n \geq 2$, và giả sử $a_0 = 3, a_1 = 5$. Tìm a_2 và a_3?
- **Lời giải**. Từ hệ thức truy hồi ta có:

$a_2 = a_1 - a_0 = 5 - 3 = 2,$

$a_3 = a_2 - a_1 = 2 - 5 = -3.$

Mô hình hóa hệ thức truy hồi

- Sử dụng hệ thức truy hồi, ta có thể mô hình hóa được lớp rất rộng trong thực tế. Mỗi bài toán cụ thể ta có một phương pháp mô hình hóa khác nhau. Dưới đây là một số ví dụ điển hình.
- **Ví dụ 15**: Bài toán dân số. Giả sử năm 1995, dân số thế giới là 7 tỉ người. Mỗi năm, dân số thế giới tăng 3%. Đến năm 2020, dân số thế giới là bao nhiêu người?

Ví dụ 15

- **Lời giải**: Gọi dân số thế giới sau n năm là P_n. Khi đó, dân số năm thứ n bằng 1.03 dân số thế giới năm trước đó. Từ đó ta có công thức truy hồi cho dãy $\{P_n\}$ như sau.

$P_n = 1.03P_{n-1}$, với $n \geq 1$ và $P_0 = 7$.

Để tính P_n ta có thể sử dụng phương pháp lặp như sau:

$$P_1 = 1.03\, P_0 = 1.03.7$$
$$P_2 = 1.03. P_1 = (1.03)^2 .7$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$P_n = 1.03P_{n-1} = (1.03)^n .7$$

Từ đó ta có $P_{25} = (1.03)^{25}.7$

Ví dụ 16

- **Bài toán Lãi kép**: Một người gửi X đô la vào tài khoản của mình tại ngân hàng với lãi suất kép 11% một năm. Hỏi sau 30 năm người đó có bao nhiêu tiền trong tài khoản?

Ví dụ 17

- **Họ nhà thỏ và số Fibonacci**: Một cặp thỏ (một con đực và một con cái) được nhốt trên một hòn đảo. Giả sử một cặp thỏ chưa sinh sản được trước khi đầy hai tháng tuổi. Từ khi chúng đầy hai tháng tuổi, mỗi tháng chúng đẻ được một cặp thỏ. Hãy tìm công thức truy hồi tính số cặp thỏ sau n tháng?

Ví dụ 17

Giải: Gọi f_n là số cặp thỏ sau n tháng. Ta sẽ chỉ ra, f_n $(n = 0,1,2,\ldots)$ chính là dãy số Fibonacci?

Vì sau hai tháng một cặp thỏ mới sinh sản được, do vậy số thỏ của tháng hiện tại bằng số thỏ của tháng trước f_{n-1} cộng với số thỏ mới đẻ là f_{n-2}.

Do vậy, dãy $\{f_n\}$ thỏa mãn hệ thức

$$f_n = f_{n-1} + f_{n-2};\ n \geq 2 \text{ với } f_0 = 0, f_1 = 1.$$

Số tháng	Số cặp sinh sản	Số cặp thỏ con	Tổng số cặp thỏ
1	0	1	1
2	0	1	1
3	1	1	2
4	1	2	3
5	2	3	5
6	3	5	8

Ví dụ 18

- **Bài toán tháp Hà Nội**: Có ba cọc (a), (b), (c). Trên cọc (a) có n $(n = 64)$ chiếc đĩa có đường kính giảm dần từ dưới lên. Cần phải dịch chuyển chồng đĩa từ cọc (a) sang cọc (c) tuân thủ theo qui tắc: mỗi lần chỉ được phép di chuyển một đĩa và chỉ được xếp các đĩa có đường kính nhỏ hơn lên đĩa có đường kính hơn. Trong quá trình dịch chuyển được phép sử dụng cọc (b) làm cọc trung gian. Bài toán đặt ra là, tìm số lần dịch chuyển ít nhất cho bài toán tháp Hà Nội?

Ví dụ 18

Lời giải. Gọi H_n là số lần dịch chuyển đĩa ít nhất từ cọc (a) sang cọc (c). Ta xây dựng công thức đệ qui để tính H_n.

Rõ ràng, $H_1 = 1$. Giả sử $n \geq 2$. Khi đó việc di chuyển được thực hiện thông qua các bước:

(i) Chuyển $(n-1)$ đĩa từ cọc (a) đến cọc (b) sử dụng cọc (c) làm cọc trung gian;

(ii) Chuyển 1 đĩa có đường kính lớn nhất từ cọc (a) đến cọc (c);

(iii)Chuyển $(n-1)$ đĩa từ cọc (b) đến cọc (c) sử dụng cọc (a) làm cọc trung gian;

Bước (i), (iii) ta cần giải bài toán Tháp Hà Nội với với $(n-1)$ đĩa. Bước (ii) cần dịch chuyển 1 lần. Do vậy, dãy $\{H_n\}$ thỏa mãn hệ thức truy hồi:

$$H_n = 2H_{n-1} + 1.$$

Ta có thể dùng phương pháp lặp để giải hệ thức truy hồi trên.

$$H_n = 2^n - 1$$

Nếu lấy $n = 64$ khi đó ta cần trên 500 tỉ năm mới có thể giải xong bài toán tháp Hà Nội!!!!!!!!!!!!!

Ví dụ 19

- Gọi a_n là số xâu nhị phân độ dài n không có 2 số 0 liên tiếp. Xây dựng công thức truy hồi cho a_n và tính a_6.

Ví dụ 19

- Gọi a_n là số xâu nhị phân độ dài n không có 2 số 0 liên tiếp. Xây dựng công thức truy hồi cho a_n và tính a_6.

Gợi ý: Xét với $n \geq 3$.

Ta chia làm 2 trường hợp

1) Xâu nhị phân độ dài n kết thúc bằng số 1 thỏa mãn yêu cầu bài ra
2) Xâu nhị phân độ dài n kết thúc bằng số 0 thỏa mãn yêu cầu bài ra, suy ra số thứ $(n-1)$ phải là 1.

Dễ thấy trong trường hợp 1) có a_{n-1} xâu thỏa mãn

Trường hợp 2) có a_{n-2} xâu thỏa mãn

Vậy $a_n = a_{n-1} + a_{n-2}$

Ví dụ 20

- **Tính số từ mã**: Một hệ máy tính coi một xâu các chữ số hệ thập phân là một từ mã hợp lệ nếu nó chứa một số chẵn chữ số 0. Ví dụ từ 168304073 là hợp lệ, từ 103203044 là không hợp lệ. Hãy tìm hệ thức truy hồi cho các từ mã hợp lệ có độ dài n?

Ví dụ 20

- **Tính số từ mã**: Một hệ máy tính coi một xâu các chữ số hệ thập phân là một từ mã hợp lệ nếu nó chứa một số chẵn chữ số 0. Ví dụ từ 168304073 là hợp lệ, từ 103203044 là không hợp lệ. Hãy tìm hệ thức truy hồi cho các từ mã hợp lệ có độ dài n?
- **Gợi ý**: Gọi số từ mã hợp lệ độ dài n là a_n

Xét với $n \geq 2$, Ta chia làm 2 trường hợp

1) Xâu $(n-1)$ chữ số đầu tiên là từ mã hợp lệ, suy ra chữ số cuối cùng khác 0. Vậy có $9a_{n-1}$
2) Xâu $(n-1)$ chữ số đầu tiên không là từ mã hợp lệ, suy ra chữ số cuối cùng là 0. Vậy có $(10^{n-1} - a_{n-1})$

Vậy $a_n = 9a_{n-1} + (10^{n-1} - a_{n-1}) = 8a_{n-1} + 10^{n-1}$

Bài tập

a) Hãy tìm hệ thức truy hồi và điều kiện đầu để tính số các xâu nhị phân có độ dài n có ít nhất một dãy hai số 0 liên tiếp?
b) Hãy tìm hệ thức truy hồi và điều kiện đầu tìm số các xâu nhị phân có độ dài n không có dãy ba số 1 liên tiếp?
c) Hãy tìm hệ thức truy hồi và điều kiện đầu tìm số các xâu nhị phân có độ dài n không có dãy bốn số 1 liên tiếp?
d) Hãy tìm hệ thức truy hồi và điều kiện đầu tìm số các xâu nhị phân có độ dài n có ít nhất một dãy ba số 1 liên tiếp?
e) Hãy tìm hệ thức truy hồi và điều kiện đầu tìm số các xâu nhị phân có độ dài n có ít nhất một dãy bốn số 1 liên tiếp?
f) Hãy tìm hệ thức truy hồi và điều kiện đầu tìm số các xâu nhị phân có độ dài n không có dãy k số 1 liên tiếp?
g) Hãy tìm hệ thức truy hồi và điều kiện đầu tìm số các xâu nhị phân có độ dài n có ít nhất một dãy k số 1 liên tiếp?

Phương pháp lặp giải hệ thức truy hồi

- **Phương pháp**: Lặp đến khi gặp điều kiện đầu trong các công thức truy hồi.

- **Bài tập**: Hãy tìm nghiệm của công thức truy hồi với điều kiện đầu dưới đây:
  - a) $a_n = a_{n-1} + 2$ với $a_0 = 3$.
  - b) $a_n = a_{n-1} + n$ với $a_0 = 1$.
  - c) $a_n = a_{n-1} + 2n + 3$ với $a_0 = 4$.
  - d) $a_n = a_{n-1} + 2^n$ với $a_0 = 1$.
  - e) $a_n = a_{n-1} - 2n - 3$ với $a_0 = 4$.

Hệ thức truy hồi tuyến tính thuần nhất

- **Định nghĩa**: Một hệ thức truy hồi tuyến tính thuần nhất bậc k với hệ số hằng số là hệ thức truy hồi có dạng:

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{n-k} \qquad (1)$$

trong đó, $c_1, c_2, \dots, ck$ là các hằng số và $c_k \neq 0$.

Ta cần tìm công thức trực tiếp tính số hạng a_n thõa mãn điều kiện (1).

Dãy số $\{a_n\}$ thỏa mãn điều kiện (1) sẽ được xác định duy nhất nếu nó thỏa mãn các điều kiện ban đầu như sau:

$$a_0 = C_0, a_1 = C_1, \dots, a_{k-1} = C_{k-1} \qquad (2)$$

trong đó C_0,..., C_{k-1} là các hằng số.

Ví dụ

- Hệ thức truy hồi $P_n = (1.11)P_{n-1}$ là hệ thức truy hồi tuyến tính thuần nhất bậc 1
- Hệ thức truy hồi $f_n = f_{n-1} + f_{n-2}$ là hệ thức truy hồi tuyến tính thuần nhất bậc 2
- Hệ thức truy hồi $a_n = a_{n-5}$ là hệ thức truy hồi tuyến tính thuần nhất bậc 5
- Hệ thức truy hồi $a_n = a_{n-1} + (a_{n-2})^2$ là không tuyến tính
- Hệ thức truy hồi $H_n = 2H_{n-1} + 1$ là không thuần nhất
- Hệ thức $B_n = nB_{n-1}$ không có hệ số hằng số

Phương pháp giải

- Phương pháp chung để giải các hệ thức truy hồi tuyến tính thuần nhất là tìm nghiệm dưới dạng $a_n = r^n$, trong đó r là hằng số. Chú ý rằng, $a_n = r^n$ là nghiệm của hệ thức truy hồi khi và chỉ khi

$$r^n = c_1 r^{n-1} + c_2 r^{n-2} + \cdots + \quad (3)$$
$$c_k r^{n-k} \quad (4)$$
$$r^k - c_1 r^{k-1} - c_2 r^{k-2} - \cdots - c_k = 0$$

Phương trình đặc trưng

Trường hợp nghiệm phân biệt

- **Định lý**: Cho c_1, c_2 là hai số thực. Giả sử phương trình đặc trưng

$$r^2 - c_1 r - c_2 = 0$$

có hai nghiệm phân biệt r_1, r_2. Khi đó, dãy $\{a_n\}$ là nghiệm của hệ thức truy hồi

$$a_n = c_1 a_{n-1} + c_2 a_{n-2}$$

khi và chỉ khi

$$a_n = \alpha_1 r_1^n + \alpha_2 r_2^n$$

Trong đó α_1 và α_2 là các hằng số.

Để tìm α_1 và α_2 ta sử dụng các điều kiện ban đầu.

Ví dụ 21

- **Bài toán**: Tìm nghiệm của hệ thức truy hồi

$$a_n = a_{n-1} + 2a_{n-2}$$

với $a_0 = 2, a_1 = 7$.

Ví dụ 21

- **Bài toán**: Tìm nghiệm của hệ thức truy hồi

$$a_n = a_{n-1} + 2a_{n-2}$$

với $a_0 = 2, a_1 = 7$.

Giải:

Bước 1: Tìm nghiệm của phương trình đặc trưng

$$r^2 - r - 2 = 0 \Leftrightarrow r_1 = 2, r_2 = -1.$$

Bước 2: Xây dựng công thức tổng quát cho $\{a_n\}$

$$a_n = \alpha_1 2^n + \alpha_2 (-1)^n$$

Bước 3: Xác định các hằng số dựa trên điều kiện ban đầu

$$\begin{cases} \alpha_1 + \alpha_2 = 2 \\ 2\alpha_1 - \alpha_2 = 7 \end{cases} \Leftrightarrow \begin{cases} \alpha_1 = 3 \\ \alpha_2 = -1 \end{cases}$$

Bước 4: Hoàn chỉnh nghiệm

$$a_n = 3.2^n - (-1)^n$$

Bài tập 1

- Tìm nghiệm của hệ thức truy hồi $F_n = F_{n-1} + F_{n-2}$, $n \geq 2$, với điều kiện ban đầu $F_0 = 0, F_1 = 1$.

Bài tập 2

- Tìm nghiệm của các hệ thức truy hồi với điều kiện đầu dưới đây
  1) $a_n = a_{n-1} + 6a_{n-2}$ với $n \geq 2$ và $a_0 = 3, a_1 = 6$.
  2) $a_n = 7a_{n-1} - 10a_{n-2}$ với $n \geq 2$ và $a_0 = 2, a_1 = 1$.
  3) $a_n = 13a_{n-1} - 22a_{n-2}$ với $n \geq 2$ và $a_0 = 3, a_1 = 15$.
  4) $a_n = -13a_{n-1} - 22a_{n-2}$ với $n \geq 2$ và $a_0 = 3, a_1 = 15$.

Trường hợp nghiệm kép

- **Định lý**: Cho c_1, c_2 là hai số thực. Giả sử phương trình đặc trưng

$$r^2 - c_1 r - c_2 = 0$$

có nghiệm kép $r_0 = r_1 = r_2$. Khi đó, dãy $\{a_n\}$ là nghiệm của hệ thức truy hồi

$$a_n = c_1 a_{n-1} + c_2 a_{n-2}$$

khi và chỉ khi

$$a_n = \alpha_1 r_0^n + \alpha_2 n r_0^n$$

Trong đó α_1 và α_2 là các hằng số.

Để tìm α_1 và α_2 ta sử dụng các điều kiện ban đầu.

Ví dụ 22

- **Bài toán**: Tìm nghiệm của hệ thức truy hồi

$$a_n = 6a_{n-1} - 9a_{n-2}$$

với $a_0 = 1, a_1 = 6$.

Ví dụ 22

- **Bài toán**: Tìm nghiệm của hệ thức truy hồi

$$a_n = 6a_{n-1} - 9a_{n-2}$$

với $a_0 = 1, a_1 = 6$.

- **Giải**:

Bước 1: Tìm nghiệm của phương trình đặc trưng

$$r^2 - 6r + 9 = 0 \Leftrightarrow r_0 = r_1 = r_2 = 3.$$

Bước 2: Xây dựng công thức tổng quát cho $\{a_n\}$

$$a_n = \alpha_1 3^n + \alpha_2 n 3^n$$

Bước 3: Xác định các hằng số dựa trên điều kiện ban đầu

$$\begin{cases} \alpha_1 = 1 \\ 3\alpha_1 + 3\alpha_2 = 6 \end{cases} \Leftrightarrow \begin{cases} \alpha_1 = 1 \\ \alpha_2 = 1 \end{cases}$$

Bước 4: Hoàn chỉnh nghiệm

$$a_n = 3^n + n3^n$$

Bài tập

- Tìm nghiệm của các hệ thức truy hồi thỏa mãn các điều kiện ban đầu sau đây

1) $a_n = 2a_{n-1} - a_{n-2}$, với $n \geq 2$ và $a_0 = 4, a_1 = 1$.
2) $a_n = 4a_{n-1} - 4a_{n-2}$, với $n \geq 2$ và $a_0 = 6, a_1 = 8$.
3) $a_n = -4a_{n-1} - 4a_{n-2}$, với $n \geq 2$ và $a_0 = 0, a_1 = 1$.
4) $a_n = -6a_{n-1} - 9a_{n-2}$, với $n \geq 2$ và $a_0 = 3, a_1 = -3$.
5) $a_n = 14a_{n-1} - 49a_{n-2}$ với $n \geq 2$ và $a_0 = 3, a_1 = 35$.
6) $a_n = -14a_{n-1} - 49a_{n-2}$ với $n \geq 2$ và $a_0 = 3, a_1 = 35$.

Trường hợp nghiệm phức

- **Định lý**: Cho c_1, c_2 là hai số thực. Giả sử phương trình đặc trưng

$$r^2 - c_1 r - c_2 = 0$$

có hai nghiệm phức liên hợp:

$$\begin{cases} r_1 = r(\cos(\theta) + i.\sin(\theta)) \\ r_2 = r(\cos(\theta) - i.\sin(\theta)) \end{cases}$$

Khi đó, dãy $\{a_n\}$ là nghiệm của hệ thức truy hồi

$$a_n = c_1 a_{n-1} + c_2 a_{n-2}$$

khi và chỉ khi

$$a_n = r^n(\alpha_1 \cos(n\theta) + \alpha_2 \sin(n\theta))$$

Trong đó α_1 và α_2 là các hằng số.

Để tìm α_1 và α_2 ta sử dụng các điều kiện ban đầu.

Ví dụ 23

- **Bài toán**: Tìm nghiệm của hệ thức truy hồi

$$a_n = 2a_{n-1} - 4a_{n-2}$$

với $a_1 = 4, a_2 = 4$.

Ví dụ 23

Bước 1: Tìm nghiệm của phương trình đặc trưng

$$r^2 - 2r + 4 = 0 \Leftrightarrow \begin{cases} r_1 = 2(\cos\left(\frac{\pi}{3}\right) + isin\left(\frac{\pi}{3}\right)) \\ r_2 = 2(\cos\left(\frac{\pi}{3}\right) - isin\left(\frac{\pi}{3}\right)) \end{cases}$$

Bước 2: Xây dựng công thức tổng quát cho $\{a_n\}$

$$a_n = 2^n(\alpha_1 \cos\left(n\frac{\pi}{3}\right) + \alpha_2 \sin\left(n\frac{\pi}{3}\right))$$

Bước 3: Xác định các hằng số dựa trên điều kiện ban đầu

$$\begin{cases} 2\left(\frac{1}{2}\alpha_1 + \frac{\sqrt{3}}{2}\alpha_2\right) = 4 \\ 4\left(-\frac{1}{2}\alpha_1 + \frac{\sqrt{3}}{2}\alpha_2\right) = 4 \end{cases} \Leftrightarrow \alpha_1 = 1; \alpha_2 = \sqrt{3}$$

Bước 4: Hoàn chỉnh nghiệm

$$a_n = 2^n(\cos\left(n\frac{\pi}{3}\right) + \sqrt{3}\sin\left(n\frac{\pi}{3}\right))$$

Trường hợp tổng quát

- **Định lý**: Cho $c_1, c_2, \dots, c_k$ là các số thực. Giả sử phương trình đặc trưng

$$r^k - c_1 r^{k-1} - \dots - c_{k-1} r - c_k = 0$$

Có k nghiệm phân biệt $r_1, r_2, \dots, r_k$.

Khi đó, dãy $\{a_n\}$ là nghiệm của hệ thức truy hồi

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \dots + c_k a_{n-k}$$

khi và chỉ khi

$$a_n = \alpha_1 r_1^n + \alpha_2 r_2^n + \dots + \alpha_k r_k^n$$

Trong đó $\alpha_1, \dots, \alpha_k$ là các hằng số.

Để tìm $\alpha_1, \dots, \alpha_k$ ta sử dụng các điều kiện ban đầu.

Bài tập

- Tìm nghiệm của các hệ thức truy hồi với điều kiện ban đầu sau:

1) $a_n = 6a_{n-1} - 11a_{n-2} + 6a_{n-3}$, với $n \geq 3$ và $a_0 = 2, a_1 = 5, a_2 = 15$.
2) $a_n = 2a_{n-1} + a_{n-2} - 2a_{n-3}$, với $n \geq 3$ và $a_0 = 3, a_1 = 6, a_2 = 0$.
3) $a_n = 7a_{n-2} + 6a_{n-3}$, với $n \geq 3$ và $a_0 = 9, a_1 = 10, a_2 = 32$.
4) $a_n = 2a_{n-1} + 5a_{n-2} - 6a_{n-3}$, với $n \geq 3$ và $a_0 = 7, a_1 = -4, a_2 = 8$.

Nội dung

- Giới thiệu bài toán
- Các nguyên lý đếm cơ bản
- Quy về bài toán con
- Hệ thức truy hồi
- Phương pháp hàm sinh
- Bài tập

Định nghĩa và ví dụ

- Định nghĩa: Hàm sinh $g(x)$ của dãy số $\{h_n\}_n = 0,1,2,..$ Là chuỗi vô hạn

$$g(x) = h_0 + h_1 x + h_2 x^2 + \ ... = \sum_{i=0}^{+\infty} h_i x^i$$

- Hàm $g(x)$ sinh ra dãy số đã cho như là các hệ số của nó. Chú ý: Trong trường hợp dãy số là hữu hạn thì ta biến nó thành dãy vô hạn bằng cách đưa vào các phần tử 0.
- Ví dụ: Hàm $g(x) = (1+x)^n$ sinh ra dãy các hệ số tổ hợp:

$$(1+x)^n = \sum_{k=0}^{n} C_n^k x^k$$

Một số khai triển thường gặp

- $\frac{x^k}{1-x} = x^k(1 + x + x^2 + \cdots) = x^k + x^{k+1} + x^{k+2} + \cdots$
- $\frac{1-x^{k+1}}{1-x} = 1 + x + x^2 + \cdots + x^k.$
- $\frac{1}{1-x^2} = 1 + x^2 + x^4 + x^6 + \cdots$
- $\frac{x}{1-x^2} = x(1 + x^2 + x^4 + x^6 + \cdots) = x + x^3 + x^5 + x^7 + \cdots$

Ví dụ 24

- **Bài toán**: Xây dựng hàm sinh giải quyết bài toán sau:

Có bao nhiêu cách chọn ra n quả từ 4 loại quả táo, chuối, cam, đào, trong đó có một số chẵn quả táo, một số lẻ quả chuối, không quá 4 quả cam, và ít nhất 2 quả đào?

Ví dụ 24

- **Bài toán**: Xây dựng hàm sinh giải quyết bài toán sau:

Có bao nhiêu cách chọn ra n quả từ 4 loại quả táo, chuối, cam, đào, trong đó có một số chẵn quả táo, một số lẻ quả chuối, không quá 4 quả cam, và ít nhất 2 quả đào?

Gợi ý: Hàm sinh giải bài toán này như sau

$$(1+x^2+x^4+x^6+\cdots)(x+x^3+x^5+\cdots)(1+x+x^2+x^3+x^4)(x^2+x^3+x^4+\cdots)$$

Hay $g(x)=\left[\frac{1}{1-x^2}\right]\left[\frac{x}{1-x^2}\right]\left[\frac{1-x^5}{1-x}\right]\left[\frac{x^2}{1-x}\right]$

Số cách cần đếm chính là hệ số của x^n trong triển khai $g(x)$ dưới dạng chuỗi lũy thừa.

Ví dụ 24

- **Bài toán**: Xây dựng hàm sinh giải quyết bài toán sau:

Có bao nhiêu cách chọn ra n quả từ 4 loại quả táo, chuối, cam, đào, trong đó có một số chẵn quả táo, một số lẻ quả chuối, không quá 4 quả cam, và ít nhất 2 quả đào?

Gợi ý: Hàm sinh giải bài toán này như sau

$$(1 + x^2 + x^4 + x^6 + \cdots)(x + x^3 + x^5 + \cdots)(1 + x + x^2 + x^3 + x^4)(x^2 + x^3 + x^4 + \cdots)$$

Hay $g(x) = \left[\frac{1}{1-x^2}\right]\left[\frac{x}{1-x^2}\right]\left[\frac{1-x^5}{1-x}\right]\left[\frac{x^2}{1-x}\right]$

Số cách cần đếm chính là hệ số của x^n trong triển khai $g(x)$ dưới dạng chuỗi lũy thừa.

Việc tính ra đáp số cần nhiều tính toán, nhưng có thể giải quyết trên máy tính dễ dàng!

Bài tập 1

- Tìm hàm sinh cho số cách đổi n (nghìn đồng) sử dụng các loại giấy bạc mệnh giá 1 nghìn đồng, 5 nghìn đồng, 10 nghìn đồng, 50 nghìn đồng (giả thiết là ta có một số lượng không hạn chế mỗi loại giấy bạc).

Học viện Công nghệ Bưu chính Viễn thông
Khoa Công nghệ thông tin 1

Toán rời rạc 1

Bài toán liệt kê

Nguyễn Thị Trang

Nội dung

- Giới thiệu bài toán
- Phương pháp sinh
- Phương pháp quay lui
- Bài tập

Giới thiệu bài toán liệt kê

- **Bài toán đếm**: Xây dựng công thức tính nghiệm của bài toán
- **Bài toán liệt kê**: Nghiệm của bài toán là gì?
- **Phương pháp chung để giải quyết bài toán liệt kê**: Sử dụng thuật toán vét cạn xem xét tất cả các khả năng xảy ra của các cấu hình tổ hợp để từ đó đưa ra từng nghiệm của bài toán
- **Phương pháp liệt kê cần thỏa mãn 2 điều kiện**:
  - Không được lặp lại bất kỳ cấu hình nào
  - Không được bỏ sót bất kỳ cấu hình nào
- **Các bước tiến hành giải bài toán bằng máy tính:**
  - Hiểu yêu cầu của bài toán
  - Chọn cấu trúc dữ liệu biểu diễn phương án cần duyệt
  - Chọn thuật toán phù hợp với cấu trúc dữ liệu
  - Cài đặt thuật toán & thử nghiệm chương trình
  - Tối ưu chương trình

Ví dụ 1

Bài toán: Cho hình vuông gồm 25 hình vuông đơn vị. Hãy điền các số từ 0 đến 9 vào mỗi hình vuông đơn vị sao cho những điều kiện sau được thỏa mãn:

a) Đọc từ trái sang phải theo hàng ta nhận được 5 số nguyên tố có 5 chữ số;

b) Đọc từ trên xuống dưới theo cột ta nhận được 5 số nguyên tố có 5 chữ số;

c) Đọc theo hai đường chéo chính ta nhận được 2 số nguyên tố có 5 chữ số;

d) Tổng các chữ số trong mỗi số nguyên tố đều là S cho trước.

Ví dụ hình vuông dưới đây với $S = 11$.

3	5	1	1	1
5	0	0	3	3
1	0	3	4	3
1	3	4	2	1
1	3	3	1	3

Ví dụ 1

Bước 1: Tìm tập các số nguyên tố như sau

$X = \{\, x \in [10001, \ldots, 99999] \mid x \text{ là nguyên tố và tổng các chữ số là } S \}$

Bước 2: Thực hiện chiến lược vét cạn như sau:

- Lấy $x \in X$ đặt vào hàng 1 (H1): ta điền được ô vuông 1, 2, 3, 4, 5.
- Lấy $x \in X$ có số đầu tiên trùng với ô số 1 đặt vào cột 1 (C1): ta điền được ô vuông 6, 7, 8, 9.
- Lấy $x \in X$ có số đầu tiên trùng với ô số 9, số cuối cùng trùng với ô số 5 đặt vào đường chéo chính 2 (D2): ta điền được ô vuông 10, 11, 12.
- Lấy $x \in X$ có số thứ nhất và số thứ 4 trùng với ô số 6 và 12 đặt vào hàng 2 (H2): ta điền được ô vuông 13, 14, 15.
- Lấy $x \in X$ có số thứ nhất, thứ hai, thứ 4 trùng với ô số 2, 13, 10 đặt vào cột 2 (C2): ta điền được ô vuông 16, 17.
- Làm tương tự như vậy ta, cho đến khi ta điền vào hàng 5 ô số 25.
- Cuối cùng ta chỉ cần kiểm tra $D1 \in X$ và $C5 \in X$?

Ví dụ 1

3	5	1	1	1
5	0	0	3	3
1	0	3	4	3
1	3	4	2	1
1	3	3	1	3

Thứ tự điền số

1	2	3	4	5
6	13	14	12	15
7	16	11	18	19
8	10	20	22	23
9	17	21	24	25

Nội dung

- Giới thiệu bài toán
- Phương pháp sinh
- Phương pháp quay lui
- Bài tập

Thuật toán sinh (1/2)

- Thuật toán sinh được dùng để giải lớp các bài toán thỏa mãn hai điều kiện:
  - Xác định được một thứ tự trên tập các cấu hình cần liệt kê của bài toán. Biết được cấu hình đầu tiên, biết được cấu hình cuối cùng.
  - Từ một cấu hình, ta xây dựng được thuật toán sinh ra cấu hình đứng ngay sau nó theo thứ tự.

Thuật toán sinh (2/2)

Bước1 (Khởi tạo):

<Thiết lập cấu hình đầu tiên>;

Bước 2 (Bước lặp):

while (<Lặp khi cấu hình chưa phái cuối cùng>)

{

<Đưa ra cấu hình hiện tại>;

<Sinh ra cấu hình kế tiếp>;

}

<Đưa ra cấu hình cuối cùng>;

Ví dụ 2

- **Bài toán**: Liệt kê (duyệt) các xâu nhị phân có độ dài n.

Xâu $X = (x_1 x_2 \ldots x_n)$: $x_i = 0, 1; i = 1, 2, \ldots, n$ được gọi là xâu nhị phân có độ dài n. Ví dụ với $n = 4$, ta có 16 xâu nhị phân dưới đây:

STT	*X =(x₁...xₙ)*	*F(X)*	*STT*	*X =(x₁...xₙ)*	*F(X)*
1	0000	0	9	1000	8
2	0001	1	10	1001	9
3	0010	2	11	1010	10
4	0011	3	12	1011	11
5	0100	4	13	1100	12
6	0101	5	14	1101	13
7	0110	6	15	1110	14
8	0111	7	16	1111	15

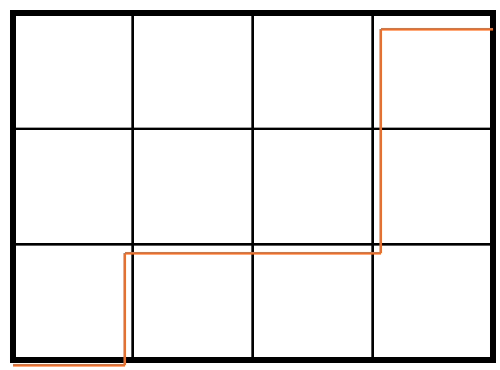
Ví dụ 2

- Thứ tự trên tập cấu hình được sắp theo giá trị của số mà cấu hình (xâu nhị phân) biểu diễn
- Cấu hình đầu tiên là xâu gồm n chữ số 0
- Cấu hình cuối cùng là xâu gồm n chữ số 1
- Thuật toán sinh cấu hình tiếp theo
  - Giả sử cấu hình hiện tại $x = x_1x_2 \dots x_n$
  - Nếu $x_i = 1$ với mọi i, thì x là cấu hình cuối cùng, thuật toán liệt kê kết thúc
  - Gọi x_k là chữ số 0 đầu tiên tính từ bên phải của x, như vậy $x = x_1x_2 \dots x_{k-1}011 \dots 1$
  - Cấu hình tiếp theo $y = y_1y_2 \dots y_n$ được tạo ra như sau
  - $y_i = x_i$ với $1 \leq i \leq k-1$, $y_i = 1 - x_i$ với $\mathrm{k} \leq i \leq n$
  - $y = x_1x_2 \dots x_{k-1}100 \dots 0$

$$\boxed{\boldsymbol{y = x + 1}}$$

Bài tập

- **Bài tập 1**. Cho một hình chữ nhật gồm $n \times m$ hình vuông đơn vị. Hãy liệt kê tất cả các đường đi từ đỉnh cuối của ô vuông cuối cùng phía bên trái đến đỉnh đầu của ô vuông trên cùng phía bên phải. Biết mỗi bước đi chỉ được phép dịch chuyển sang bên phải hoặc lên trên theo các cạnh của hình vuông đơn vị.



Bài tập

- **Bài tập 2.** Hãy liệt kê tất cả các xâu nhị phân có độ dài n sao cho mỗi xâu nhị phân có duy nhất một dãy k bít 1 liên tiếp.

- **Bài tập 3.** Hãy liệt kê tất cả các xâu nhị phân có độ dài n sao cho mỗi xâu nhị phân có duy nhất một dãy m bít 1 liên tiếp và duy nhất một dãy có k bít 0 liên tiếp.

Bài tập

- **Bài tập 4.** Chuỗi ký tự $X = (x_1, x_2, \ldots, xn)$ được gọi là chuỗi ký tự AB nếu $x_i = $ 'A' hoặc $x_i = $ 'B'. Chuỗi X được gọi là chuỗi AB bậc k nếu X tồn tại duy nhất một dãy k kí tự A liên tiếp. Hãy liệt kê tất cả các chuỗi AB bậc k.

- **Bài tập 5.** Cho dãy số $A = (a_1, a_2, \ldots, a_n)$ gồm n số tự nhiên khác nhau và số tự nhiên k. Hãy liệt kê tất cả các dãy con của dãy số A sao cho tổng các phần tử của dãy con đó đúng bằng k.

Ví dụ 3

- Liệt kê (duyệt) các tổ hợp chập k của $1, 2, \ldots, n$.

Mỗi tổ hợp chập k của $1, 2, \ldots, n$ là một tập con k phần tử khác nhau của $1, 2, \ldots, n$.

Ví dụ với $n = 5$, $k = 3$ ta sẽ có C_n^k tập con dưới đây

STT	Tập con $X = \{x_1, \ldots, x_k\}$	STT	Tập con $X = \{x_1, \ldots, x_k\}$
1	1 2 3	6	1 4 5
2	1 2 4	7	2 3 4
3	1 2 5	8	2 3 5
4	1 3 4	9	2 4 5
5	1 3 5	10	3 4 5

Ví dụ 3

- **Thứ tự tự nhiên duyệt các tổ hợp chập k**

Có thể xác định được nhiều trật tự khác nhau trên các tổ hợp. Tuy nhiên, thứ tự đơn giản nhất có thể được xác định như sau:

Ta gọi tập con $X = (x_1, x_2, \ldots, xk)$ là đứng trước tập con $Y = (y_1, y_2, \ldots, yk)$ nếu tìm được chỉ số t sao cho $x_1 = y_1$, $x_2 = y_2, \ldots, x_{t-1} = y_{t-1}, x_t < y_t$.

Ví dụ tập con $X = (1, 2, 3)$ đứng trước tập con $Y = (1, 2, 4)$ vì với $t = 3$ thì $x_1 = y_1, x_2 = y_2, x_3 < y_3$.

Tập con (cấu hình) đầu tiên là $X = (1, 2, \ldots, k)$, tập con (cấu hình) cuối cùng là $(n - k + 1, \ldots, n)$. Như vậy điều kiện 1 của thuật toán sinh được thỏa mãn.

Ví dụ 3

- Thuật toán sinh cấu hình (tổ hợp) tiếp theo
- Giả sử cấu hình hiện tại là $X = (x_1, x_2, \ldots, x_k)$
- Nếu $x_i = n - k + i$ với mọi $i = 1,2, \ldots, k$ thì X là cấu hình cuối cùng. Thuật toán duyệt kết thúc
- Gọi t là chỉ số lớn nhất (x_t là số đầu tiên từ phải sang) sao cho $x_t < n - k + t$
- Cấu hình tiếp theo $Y = (y_1, y_2, \ldots, y_k)$ được sinh ra như sau
  - $y_i = x_i$ với $i < t$,
  - $y_t = x_t + 1$,
  - $y_i = y_t + (i - t)$ với $i > t$

Bài tập

- **Bài tập 6.** Cho dãy số $A = (a_1, a_2, \ldots, a_n)$ và số tự nhiên P. Hãy liệt kê tất cả các dãy con k phần tử của dãy số A sao cho tổng các phần tử của dãy con đó đúng bằng P.

Ví dụ. $A = (5, 10, 15, 20, 25, 30, 35)$, $n = 7, k = 3, P = 50$ ta sẽ có các dãy con sau :

(5, 10, 35),
(5, 20, 25),
(10, 15, 25),
...

- **Bài tập 7.** Cho dãy số $A = (a_1, a_2, \ldots, a_n)$. Hãy liệt kê tất cả các dãy con k phần tử tăng dần tự nhiên của dãy số A.

Ví dụ. $A = (1, 3, 2, 4, 5), n = 5, k = 3$ ta có các dãy con tăng dần tự nhiên như sau :

(1, 3, 4),
(1, 3, 5),
(1, 2, 4),
...

Ví dụ 4

- Liệt kê (duyệt) các hoán vị của $1,2,\ldots,n$.

Mỗi hoán vị của $1,2,\ldots,n$ là một cách xếp có tính đến thứ tự của $1,2,\ldots,n$. Số các hoán vị là $n!$. Ví dụ với $n = 3$ ta có 6 hoán vị dưới đây.

STT	Hoán vị $X = (x_1, \ldots, x_n)$
1	1 2 3
2	1 3 2
3	2 1 3
4	2 3 1
5	3 1 2
6	3 2 1

Ví dụ 4

- **Thứ tự tự nhiên duyệt hoán vị**

Có thể xác định được nhiều trật tự khác nhau trên các hoán vị. Tuy nhiên, thứ tự đơn giản nhất có thể được xác định như sau. Hoán vị $X = (x_1, x_2, \ldots, x_n)$ được gọi là đứng sau hoán vị $Y = (y_1, y_2, \ldots, y_n)$ nếu tồn tại chỉ số k sao cho

$$x_1 = y_1, x_2 = y_2, \ldots, x_{k-1} = y_{k-1}, x_k < y_k.$$

Ví dụ hoán vị $X = (1, 2, 3)$ được gọi là đứng sau hoán vị $Y = (1, 3, 2)$ vì tồn tại $k = 2$ để $x_1 = y_1$, và $x_2 < y_2$.

Cấu hình đầu tiên là $(1,2, \ldots, n)$

Cấu hình cuối cùng là $(n, n-1, \ldots, 1)$

Ví dụ 4

- Thuật toán sinh cấu hình (hoán vị) tiếp theo
- Giả sử cấu hình hiện tại là $X = (x_1, x_2, \ldots, x_n)$
- Nếu $x_{i-1} > x_i$ với mọi i, thì X là cấu hình cuối cùng. Thuật toán sinh kết thúc.
- Gọi t là chỉ số lớn nhất (chỉ số đầu tiên từ bên phải) sao cho $x_{t-1} < x_t$.
- Cấu hình tiếp theo $Y = (y_1, y_2, \ldots, y_n)$ được sinh ra như sau
  - $y_i = x_i$ với $i \leq t - 2$
  - y_{t-1} bằng phần tử nhỏ nhất trong tập $x_t,\ldots,x_n$ và lớn hơn x_{t-1} (ký hiệu là a)
  - $y_t,\ldots,y_n$ là dãy sắp xếp tăng dần gồm các số trong tập $\{x_{t-1}, x_t, \ldots, x_n\} \backslash \{a\}$

Bài tập

- **Bài tập 8.** Một dãy số tự nhiên bất kỳ $A_n = \{a_1, a_2, \ldots, a_n\}$ được gọi là một đường nguyên tố bậc k nếu tổng k phần tử liên tiếp bất kỳ của dãy số A_n là một số nguyên tố $(k \le n)$. Ví dụ dãy số $A_n = \{3, 27, 7, 9, 15\}$ là một đường nguyên tố bậc 3. Cho dãy số A_n. Hãy liệt kê tất cả các đường nguyên tố bậc k có thể có được tạo ra bằng cách tráo đổi các phần tử khác nhau của dãy số A_n.

- Ví dụ với dãy $A = (3, 7, 9, 15, 27)$ ta sẽ thành lập được 4 dãy nguyên tố thuần nhất bậc 3 như dưới đây:

3	27	7	9	15
15	9	7	3	27
15	9	7	27	3
27	3	7	9	15

Nội dung

- Giới thiệu bài toán
- Phương pháp sinh
- Phương pháp quay lui
- Bài tập

Thuật toán quay lui (1/2)

- Giả sử ta cần xác định bộ $X = (x_1, x_2, \ldots, x_n)$ thỏa mãn một số ràng buộc nào đó. Ứng với mỗi thành phần x_i ta có n_i khả năng cần lựa chọn.
- Ứng với mỗi khả năng j trong n_i dành cho thành phần x_i ta cần thực hiện:
  - Kiểm tra xem khả năng j có được chấp thuận cho thành phần x_i hay không?
    - Nếu khả năng j được chấp thuận thì nếu i là thành phần cuối cùng $(i = n)$ ta ghi nhận nghiệm của bài toán. Nếu i chưa phải cuối cùng ta xác định thành phần thứ $i + 1$.
    - Nếu không có khả năng j nào được chấp thuận cho thành phần x_i thì ta quay lại bước trước đó $(i - 1)$ để thử lại các khả năng khác.

Thuật toán quay lui (2/2)

```
Back_Track (int i ) {
    for ( j =<Khả năng 1>; j <=n_i;  j++ ){
        if (<chấp thuận khả năng j>) {
            X[i] = <khả năng j>;
            if ( i ==n)
                Result();
            else
                Back_Track(i+1);
        }
    }
```

Ví dụ 5

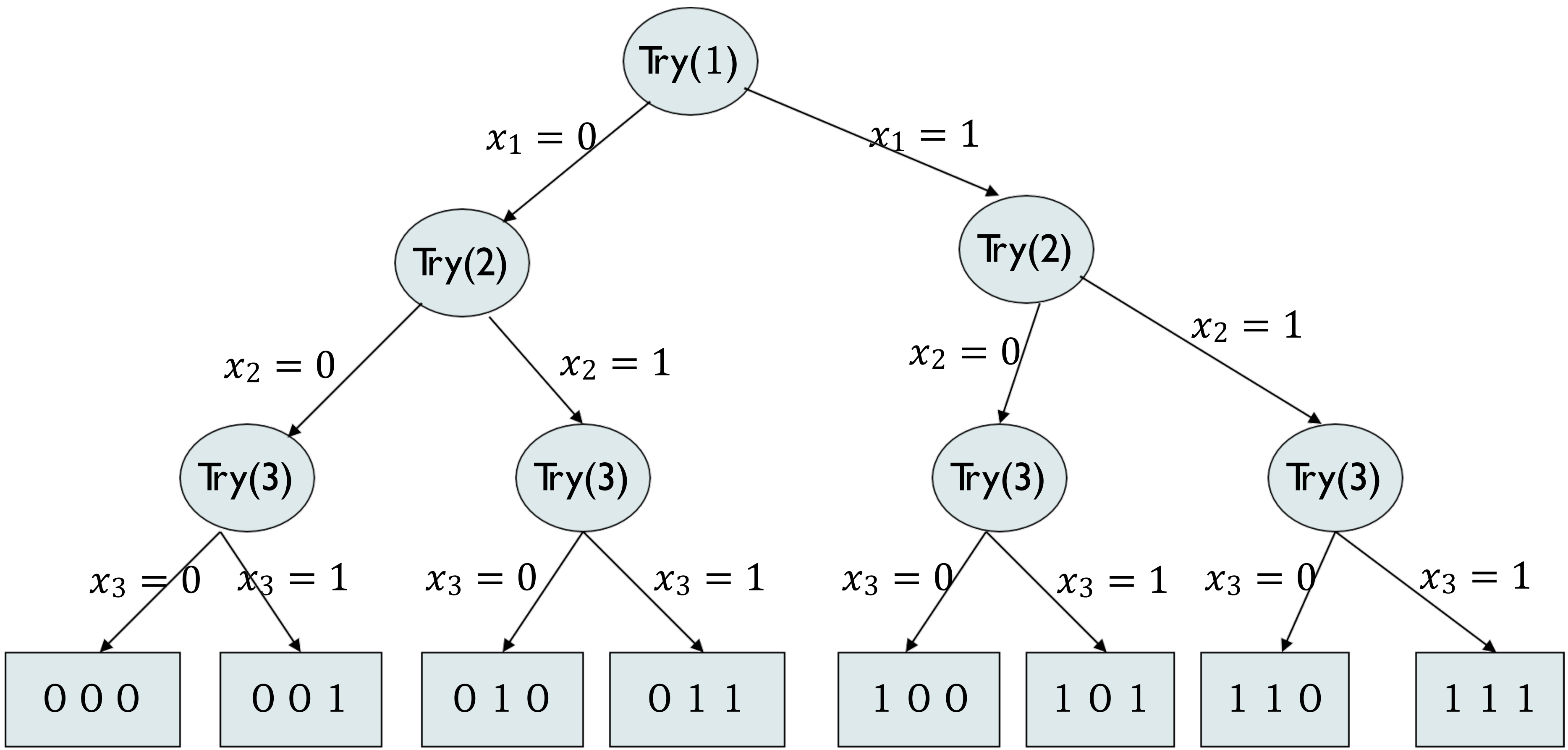
- Liệt kê (duyệt) các xâu nhị phân có độ dài n.

Xâu $X = (x_1x_2\ldots x_n)$: $x_i = 0, 1; i = 1, 2, \ldots, n$ được gọi là xâu nhị phân có độ dài n.

```
void Try ( int i ) {
        for (int j =0; j<=1; j++){
                X[i] = j;
                if ( i ==n)
                        Result();
                else
                        Try (i+1);
        }
}
```

Khi đó, để duyệt các xâu nhị phân có độ dài n ta chỉ cần gọi đến thủ tục Try(1).

Ví dụ 5


Try(1)
$x_1 = 0$
$x_1 = 1$
Try(2)
Try(2)
$x_2 = 0$
$x_2 = 1$
$x_2 = 0$
$x_2 = 1$
Try(3)
Try(3)
Try(3)
Try(3)
$x_3 = 0$
$x_3 = 1$
$x_3 = 0$
$x_3 = 1$
$x_3 = 0$
$x_3 = 1$
$x_3 = 0$
$x_3 = 1$
0 0 0
0 0 1
0 1 0
0 1 1
1 0 0
1 0 1
1 1 0
1 1 1

Bài tập

- **Bài tập 9**. Sử dụng thuật toán quay lui, hãy liệt kê tất cả các phần tử của tập:

$$D = \left\{ X = (x_1, x_2, .., x_n) : \sum_{i=1}^{n} a_i x_i \le W \wedge \sum_{i=1}^{n} c_i x_i = K \right\}.$$

Trong đó, $x_i = 0, 1; c_i, a_i \in Z^+$;

$n \le 100, W \le 32000;\ K \le 32000.$

Ví dụ 6

- Liệt kê (duyệt) các tổ hợp chập k của $1, 2, \ldots, n$.

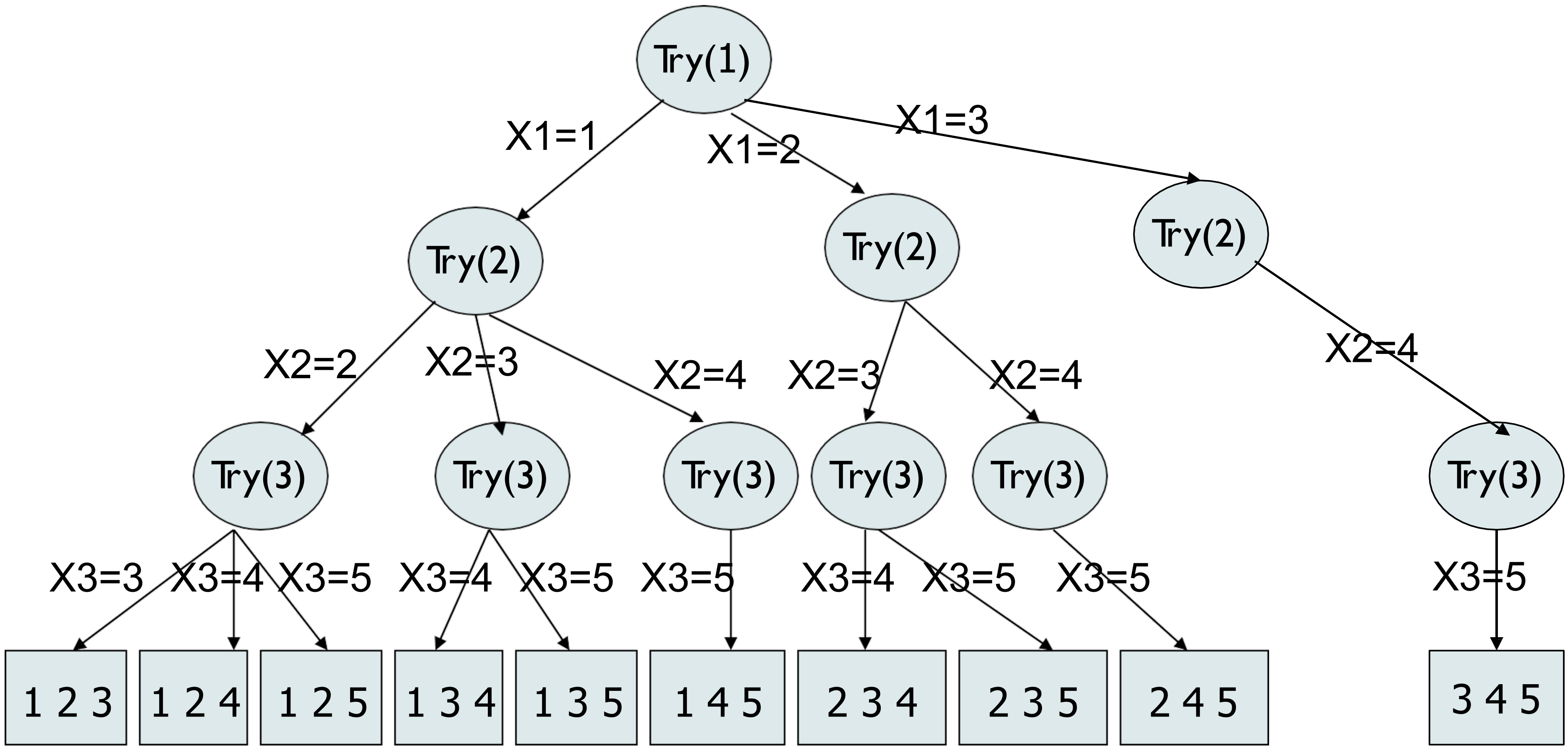
Mỗi tổ hợp chập k của $1, 2, \ldots, n$ là một tập con k phần tử khác nhau của $1, 2, \ldots, n$.

```
void Try ( int i ) {
        for (int j =X[i-1]+1; j<=n-k+ i; j++){
            X[i] = j;
            if ( i ==k)
                Result();
            else
                Try (i+1);
        }
}
```

Coi $X[0] = 0$.
Ta cần gán giá trị cho $X[1], \ldots, X[k]$

Khi đó, để duyệt các tập con k phần tử của $1, 2, \ldots, n$ ta chỉ cần gọi đến thủ tục Try(1).

Ví dụ 6



Try(1)
X1=1
X1=2
X1=3
Try(2)
Try(2)
Try(2)
X2=2
X2=3
X2=4
X2=3
X2=4
X2=4
Try(3)
Try(3)
Try(3)
Try(3)
Try(3)
Try(3)
X3=3
X3=4
X3=5
X3=4
X3=5
X3=5
X3=4
X3=5
X3=5
X3=5
1 2 3
1 2 4
1 2 5
1 3 4
1 3 5
1 4 5
2 3 4
2 3 5
2 4 5
3 4 5

Bài tập

- **Bài tập 10**. Sử dụng thuật toán quay lui, hãy liệt kê tất cả các phần tử của tập:

$$D = \left\{ X = (x_1, x_2, .., x_n) : \sum_{i=1}^{n} x_i = K \wedge \sum_{i=1}^{n} a_i x_i = S \right\}.$$

Trong đó, $x_i = 0, 1$; $a_i \in Z^{+}$;

$n \leq 100, K \leq 100;\ S \leq 32000.$

Ví dụ 7

- Sử dụng phương pháp quay lui liệt kê (duyệt) các hoán vị của $1,2,\dots,n$.

Mỗi hoán vị $X = (x_1, x_2, \dots, xn)$ là bộ có tính đến thứ tự của $1,2,\dots,n$. Mỗi $x_i \in X$ có n lựa chọn. Khi $x_i = j$ được lựa chọn thì giá trị này sẽ không được chấp thuận cho các thành phần còn lại.

Để ghi nhận điều này, ta sử dụng mảng $unused[]$ gồm n phần tử.

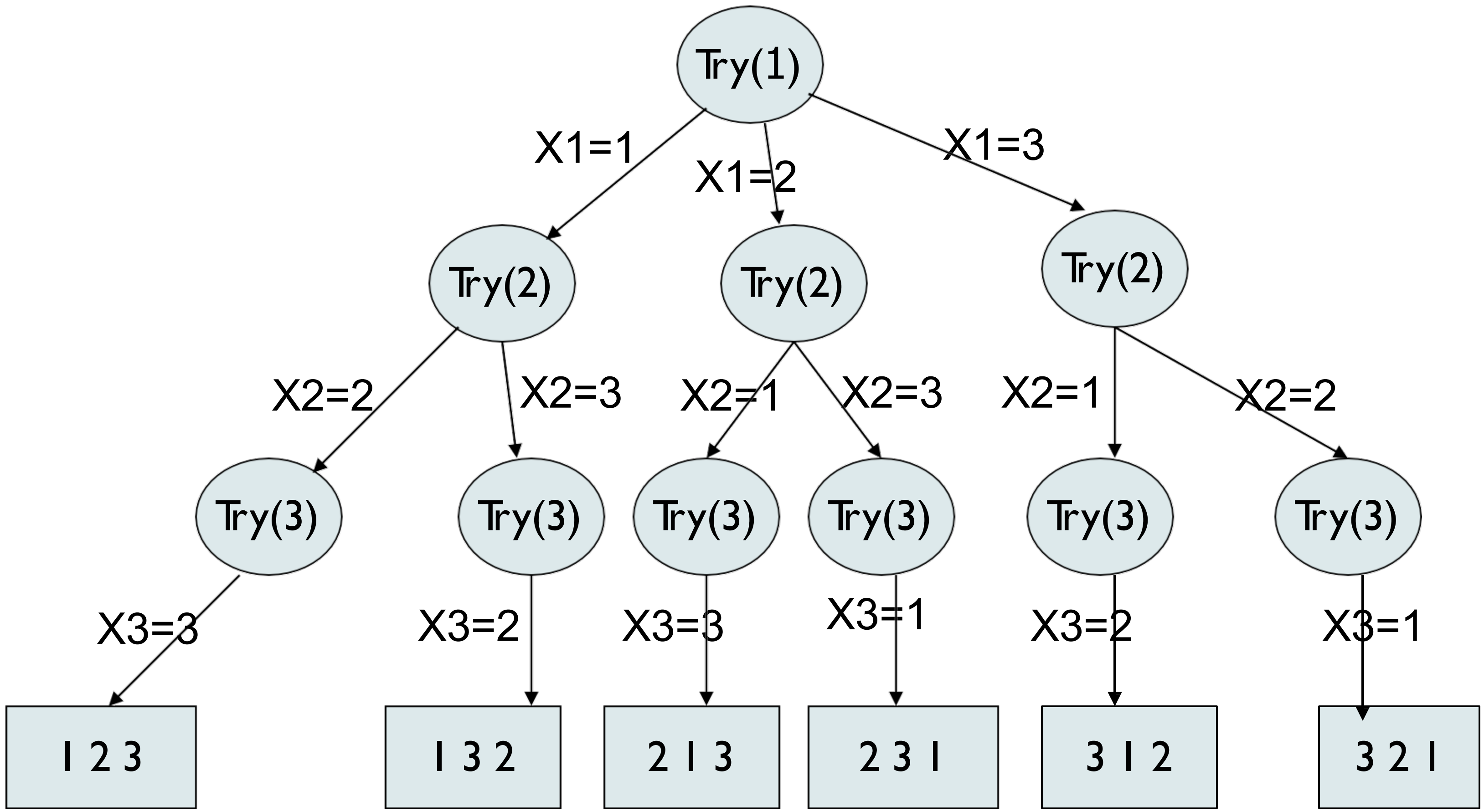
- Nếu $unused[i] = True$ điều đó có nghĩa giá trị i được chấp thuận
- $unused[i] = False$ tương ứng với giá trị i không được phép sử dụng

Ví dụ 7

```
void Try ( int i ) {
        for (int j =1; j<=n; j++){
                if (unused[j] ) {
                        X[i] = j;
                        unused[j] = false;
                        if ( i ==n)
                                Result();
                        else
                                Try (i+1);
                        unused[j] = true;
                }
        }
}
```

Khi đó, để duyệt các hoán vị của $1, 2, \dots, n$ ta chỉ cần gọi đến thủ tục Try(1).

Ví dụ 7


Try(1)
X1=1
X1=2
X1=3
Try(2)
Try(2)
Try(2)
X2=2
X2=3
X2=1
X2=3
X2=1
X2=2
Try(3)
Try(3)
Try(3)
Try(3)
Try(3)
Try(3)
X3=3
X3=2
X3=3
X3=1
X3=2
X3=1
1 2 3
1 3 2
2 1 3
2 3 1
3 1 2
3 2 1

Ví dụ 8

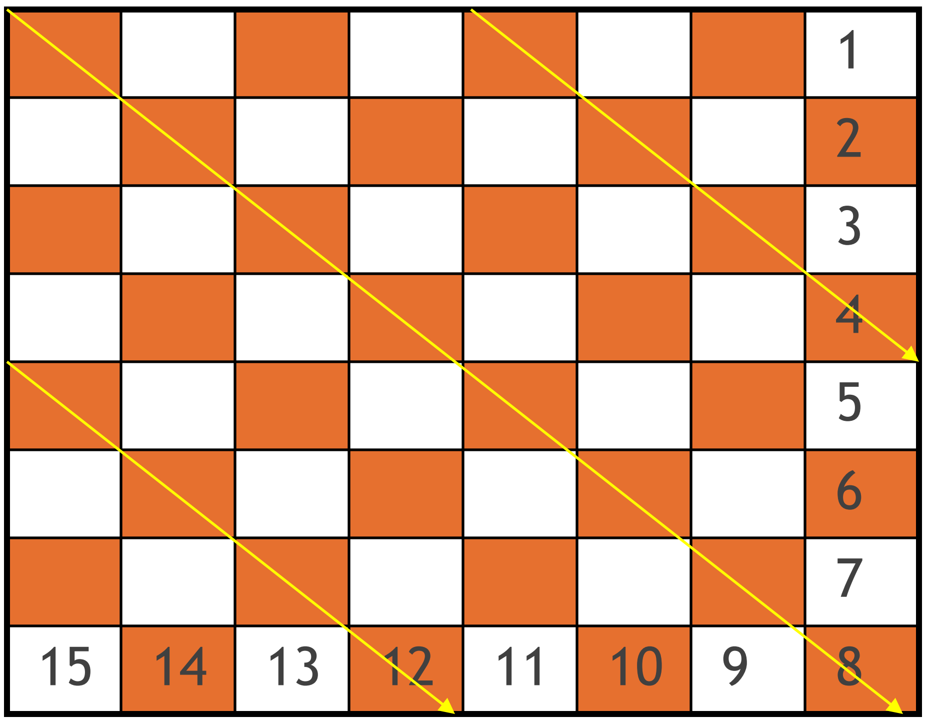
- **Bài toán n quân hậu**. Trên bàn cờ kích cỡ $n \times n$, hãy đặt n quân hậu mỗi quân trên 1 hàng sao cho tất cả các quân hậu đều không ăn được lẫn nhau.

Gọi $X = (x_1, x_2, \ldots, xn)$ là một nghiệm của bài toán. Khi đó, $x_i = j$ được hiểu là quân hậu hàng thứ i đặt ở cột j. Để các quân hậu khác không thể ăn được, quân hậu thứ i cần không được lấy trùng với bất kỳ cột nào, không được cùng đường chéo xuôi, không được cùng trên đường chéo ngược.

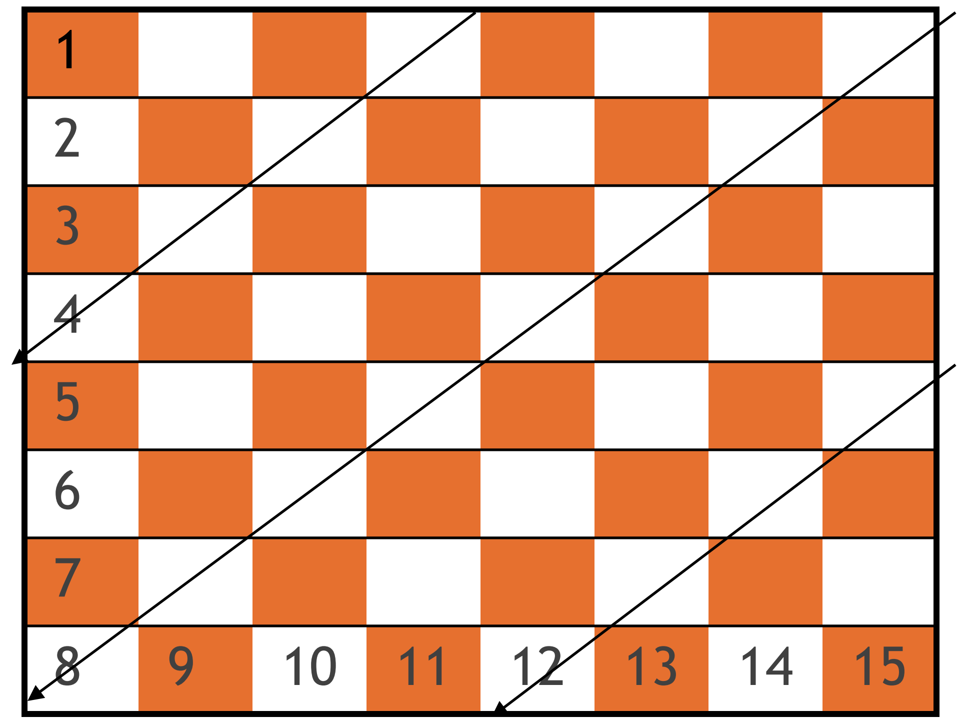
Ta có n cột $A = (a_1, \ldots, a_n)$, có $Xuoi[2 * n - 1]$ đường chéo xuôi, $Nguoc[2 * n - 1]$ đường chéo ngược.

Ví dụ 8

Đường chéo xuôi: $Xuoi\,[i - j + n]$



Đường chéo ngược: $Nguoc\,[i + j - 1]$



Ví dụ 8

```
void Try (int i){
        for(int j=1; j<=n; j++){
                if( A[j] && Xuoi[ i - j + n ] && Nguoc[i + j -1]){
                        X[i] =j;
                        A[j]=false;
                        Xuoi[ i - j + n]=false;
                        Nguoc[ i + j - 1]=false;
                        if(i==n) Result();
                        else Try(i+1);
                        A[j] = true;
                        Xuoi[ i - j + n] = true;
                        Nguoc[ i + j - 1] = true;
                }
        }
}
```

Khi đó, để giải bài toán quân hậu ta chỉ cần gọi đến thủ tục Try(1).

Bài tập

I. Sử dụng thuật toán sinh, viết chương trình giải các bài tập dưới đây:

1. Liệt kê các xâu nhị phân có độ dài n.
2. Liệt kê các tập con k phần tử của $1, 2, \ldots, n$.
3. Liệt kê các hoán vị của $1, 2, \ldots, n$.
4. Liệt kê các cách chia số tự nhiên n thành tổng các số nhỏ hơn n.
5. Liệt kê tất cả các xâu nhị phân độ dài n có duy nhất một dãy k bít 0 liên tiếp và duy nhất một dãy m bít 1liền tiếp.
6. Liệt kê các dãy con của dãy số A_n sao cho tổng các phần tử của dãy con đó đúng bằng k.

1.7. Liệt kê tất các các dãy con k phần tử của dãy số A_n sao cho tổng các phần tử của dãy co đó đúng bằng B.

8.Liệt kê tất cả các cách chọn trên mỗi hàng, mỗi cột khác nhau các phần tử của ma trận vuông A cấp n sao cho tổng các phần tử đó đúng bằng K.

9.Liệt kê tất cả các dãy số nguyên tố thuần nhất bậc k của dãy số A_n bằng cách tráo đổi nội dung các phần tử của dãy số A_n.

10. Giải bài toán n quân hậu.

Bài tập

- II. Sử dụng thuật toán quay lui, viết chương trình giải các bài tập dưới
- đây:

1. Liệt kê các xâu nhị phân có độ dài n.
2. Liệt kê các tập con k phần tử của $1, 2, \ldots, n$.
3. Liệt kê các hoán vị của $1, 2, \ldots, n$.
4. Liệt kê các cách chia số tự nhiên n thành tổng các số nhỏ hơn n.
5. Liệt kê tất cả các xâu nhị phân độ dài n có duy nhất một dãy k bít 0 liên tiếp

- và duy nhất một dãy m bít 1liên tiếp.

6. Liệt kê các dãy con của dãy số A_n sao cho tổng các phần tử của dãy con đó đúng bằng k.

2.7. Liệt kê tất các các dãy con k phần tử của dãy số A_n sao cho tổng các phần tử của dãy co đó đúng bằng B.

8.Liệt kê tất cả các cách chọn trên mỗi hàng, mỗi cột khác nhau các phần tử của ma trận vuông A cấp n sao cho tổng các phần tử đó đúng bằng K.

9.Liệt kê tất cả các dãy số nguyên tố thuần nhất bậc k của dãy số A_n bằng cách tráo đổi nội dung các phần tử của dãy số A_n.

10. Giải bài toán n quân hậu.