🎓 Handout: Bài toán đếm trong Nguyên lý dirichlet & Các cấu hình tổ hợp

Sau khi hoàn thành hai chủ đề này, sinh viên có thể:

- ✅ Hiểu và phát biểu được nguyên lý Dirichlet ở cả dạng cơ bản và tổng quát
- ✅ Vận dụng nguyên lý Dirichlet để giải quyết các bài toán chứng minh tồn tại trong phân phối, lập lịch, hệ thống
- ✅ Nhận biết và phân biệt chính xác các loại cấu hình tổ hợp cơ bản (hoán vị, chỉnh hợp, tổ hợp...) và có lặp
- ✅ Áp dụng đúng công thức tổ hợp tương ứng với ngữ cảnh bài toán
- ✅ Giải quyết các bài toán tổ hợp thường gặp trong lập trình, thuật toán, kiểm thử hệ thống, phân tích dữ liệu

PHẦN 1: LÝ THUYẾT 1

Trong chương này, chúng ta sẽ tìm hiểu hai chủ đề trọng tâm trong tổ hợp rời rạc: **Nguyên lý Dirichlet** và **Các cấu hình tổ hợp cơ bản**. Đây là những kiến thức nền tảng có tính ứng dụng cao trong khoa học máy tính, đặc biệt trong thiết kế thuật toán, kiểm thử phần mềm, bảo mật thông tin, và phân tích dữ liệu.

Nguyên lý Dirichlet, hay còn gọi là nguyên lý "lồng chim bồ câu", là một nguyên lý đơn giản nhưng rất mạnh trong việc chứng minh sự tồn tại. Sinh viên sẽ được tiếp cận nguyên lý này thông qua các phát biểu trực quan, các dạng tổng quát và ứng dụng vào các tình huống thực tiễn như va chạm trong hàm băm, phân bổ tài nguyên trong hệ thống, và kiểm tra tính duy nhất trong lập lịch.

Tiếp theo, phần "Các cấu hình tổ hợp" sẽ trang bị cho sinh viên kỹ năng nhận diện và áp dụng chính xác các công thức đếm trong tổ hợp: hoán vị, chỉnh hợp, tổ hợp và các biến thể có lặp. Sinh viên sẽ học cách phân biệt rõ khi nào cần xét thứ tự, khi nào cho phép chọn lặp, từ đó áp dụng đúng công thức tổ hợp tương ứng. Nội dung này đóng vai trò quan trọng trong việc xây dựng không gian tìm kiếm, sinh cấu hình đầu vào, tối ưu hóa truy vấn và đánh giá độ phức tạp thuật toán.

Sau khi hoàn thành nội dung bài học, sinh viên cần có khả năng phát biểu và vận dụng nguyên lý Dirichlet để chứng minh các tình huống tồn tại, đồng thời áp dụng linh hoạt các cấu hình tổ hợp trong việc giải các bài toán đếm, mô hình hóa lựa chọn và sinh dữ liệu trong môi trường công nghệ thông tin.

📚 NỘI DUNG CHÍNH

🔷 3. NGUYÊN LÝ DIRICHLET (Pigeonhole Principle)

- Trực quan – động lực tư duy

Phát biểu dạng cơ bản & tổng quát

- Các dạng biến thể – phản chứng
- Ứng dụng trong thuật toán, bảo mật, hệ thống
- Bài tập luyện tập & mở rộng

🔷 4. CÁC CẤU HÌNH TỔ HỢP (Combinatorial Configurations)

- Hoán vị, chỉnh hợp, tổ hợp
- Hoán vị lặp, tổ hợp có lặp
- Cách phân biệt và chọn đúng cấu hình
- Bảng tổng hợp – sơ đồ nhận diện
- Ứng dụng trong sinh mã, thuật toán sinh, phân tích tổ hợp

🧠 NĂNG LỰC PHÁT TRIỂN

- **Tư duy logic – cấu trúc**
- **Kỹ năng phân tích bài toán đếm**
- **Khả năng ứng dụng công thức tổ hợp linh hoạt trong CNTT**
- **Phân biệt được bản chất các tình huống có/không lặp – có/không thứ tự**
- **Chuẩn bị tốt nền tảng cho các chủ đề nâng cao: truy hồi, bài toán liệt kê, xác suất rời rạc**

🎓 A: NGUYÊN LÝ DIRICHLET (Pigeonhole Principle)

🔍 1.1 Trực quan & Động lực

Nguyên lý Dirichlet – thường được gọi là *nguyên lý lồng chim bồ câu* – là một định lý cơ bản nhưng đầy sức mạnh trong tổ hợp rời rạc. Nguyên lý này phát biểu một ý tưởng trực quan: **khi số lượng đối tượng vượt quá số lượng vị trí chứa, thì sẽ có ít nhất một vị trí chứa từ hai đối tượng trở lên**.

Hình dung một cách đơn giản: nếu có 10 đôi tất cần cất vào 9 ngăn kéo, thì chắc chắn tồn tại ít nhất một ngăn kéo chứa từ hai đôi tất trở lên. Tương tự, nếu có 13 sinh viên và chỉ 7 ngày trong tuần, thì sẽ có ít nhất hai (thậm chí ba) sinh viên trùng ngày sinh.

Mặc dù phát biểu đơn giản, nguyên lý Dirichlet thường được sử dụng để **chứng minh sự tồn tại của một hiện tượng** thay vì chỉ ra cụ thể vị trí hay đối tượng đó là gì. Đây chính là lý do nguyên lý này trở thành công cụ quen thuộc trong các bài toán về phân phối, lập lịch, an toàn dữ liệu, và trong phân tích thuật toán – những lĩnh vực có mối liên hệ chặt chẽ với ngành CNTT.

1.2 Định nghĩa chính thức

Nguyên lý Dirichlet có thể được phát biểu chính xác dưới dạng sau:

> **Nếu phân bố hơn n đối tượng vào n ngăn chứa (hộp), thì tồn tại ít nhất một hộp chứa từ hai đối tượng trở lên.**

Phiên bản mở rộng và thường được sử dụng trong các bài toán thực tế là:

> **Nếu m đối tượng được phân vào n hộp, thì tồn tại ít nhất một hộp chứa không ít hơn $\lceil \frac{m}{n} \rceil$ đối tượng.**

Ở đây, $\lceil x \rceil$ là ký hiệu chỉ phần nguyên lớn nhất không nhỏ hơn x (*ceiling function*).

Nguyên lý này không chỉ mang giá trị lý thuyết, mà còn là một công cụ suy luận tồn tại mạnh mẽ trong các chứng minh tổ hợp và thuật toán. Trong bối cảnh khoa học máy tính, nguyên lý Dirichlet thường được sử dụng để **chứng minh sự va chạm, trùng lặp, hoặc ràng buộc không thể tránh khỏi trong hệ thống phân bổ tài nguyên, mã hóa, hoặc hàm băm.**

Về bản chất, nguyên lý này làm nổi bật một giới hạn không thể vượt qua trong quá trình ánh xạ giữa hai tập hợp với số lượng phần tử chênh lệch.

1.3 Các dạng phát biểu khác

Nguyên lý Dirichlet tồn tại dưới nhiều dạng phát biểu tương đương, được điều chỉnh tùy theo mục đích và bối cảnh áp dụng. Việc nhận diện đúng dạng phù hợp sẽ giúp sinh viên xử lý linh hoạt các tình huống đếm và chứng minh trong tổ hợp và thuật toán.

a) Dạng cơ bản (Minimal Form)

> Nếu $m > n$, thì khi phân m đối tượng vào n hộp, tồn tại ít nhất một hộp chứa từ **2 đối tượng trở lên**.

Đây là dạng đơn giản nhất, thường dùng trong các bài toán chứng minh tồn tại hoặc tìm mâu thuẫn.

b) Dạng tổng quát (Generalized Form)

> Nếu m đối tượng được phân vào n hộp, thì tồn tại ít nhất một hộp chứa không ít hơn: $\lceil \frac{m}{n} \rceil$ đối tượng.

Dạng này cung cấp một đánh giá chính xác hơn về mức độ “chồng lấn”, đặc biệt hữu ích khi cần xác định ngưỡng ràng buộc trong các bài toán tối ưu.

c) Dạng phủ định (Contrapositive Form)

> Nếu mỗi hộp chứa nhiều nhất k đối tượng, thì tổng số đối tượng không vượt quá $n \cdot k$.

Dạng phủ định được sử dụng trong các chứng minh phản chứng hoặc để xác lập giới hạn tối đa trong thiết kế hệ thống.

Dạng phát biểu	Biểu thức đặc trưng	Mục đích sử dụng
Cơ bản	$m > n \rightarrow \exists$ hộp chứa ≥ 2	Chứng minh tồn tại đơn giản
Tổng quát	$m \rightarrow n \rightarrow \exists$ hộp chứa $\geq \lceil \frac{m}{n} \rceil$	Ước lượng chính xác
Phủ định	$m \leq n \cdot k \rightarrow$ mỗi hộp $\leq k$	Phân tích ngược, giới hạn trên

Ba dạng phát biểu trên không chỉ là hình thức biểu đạt khác nhau, mà còn thể hiện sự đa dạng trong tư duy toán học: từ trực quan đến định lượng, từ kết luận đến điều kiện ngược lại. Việc nắm vững và sử dụng linh hoạt các dạng này sẽ giúp sinh viên khai thác tối đa tiềm năng ứng dụng của nguyên lý Dirichlet trong các lĩnh vực thuộc CNTT.

🧪 1.4 Ví dụ minh họa

Việc hiểu và vận dụng nguyên lý Dirichlet sẽ trở nên rõ ràng hơn thông qua các ví dụ thực tiễn. Các tình huống dưới đây không chỉ giúp sinh viên củng cố khái niệm, mà còn làm rõ tính ứng dụng của nguyên lý trong đời sống và lĩnh vực công nghệ thông tin.

◈ Ví dụ 1 – Ngày sinh trùng lặp

Bài toán: Trong một hội trường có 367 người, chứng minh rằng tồn tại ít nhất hai người có cùng ngày sinh.

Phân tích: Một năm có tối đa 366 ngày (nếu tính cả năm nhuận). Áp dụng nguyên lý Dirichlet với:

- Số hộp = số ngày sinh khả dĩ = 366
- Số đối tượng = 367 người

⇒ Phải có ít nhất một ngày có từ **2 người trở lên** sinh nhật, do: $\lceil \frac{367}{366} \rceil = 2$

◈ Ví dụ 2 – Sinh viên và ngày trong tuần

Bài toán: Trong một lớp có 15 sinh viên, chứng minh rằng tồn tại ít nhất 3 người có ngày sinh rơi vào cùng một thứ trong tuần.

Phân tích:

- Số hộp = 7 (từ Thứ Hai đến Chủ Nhật)
- Số sinh viên = 15

Áp dụng Dirichlet tổng quát: $\lceil \frac{15}{7} \rceil = 3$: ➔ Tồn tại ít nhất một ngày có ≥ 3 sinh viên sinh vào ngày đó.

◈ Ví dụ 3 – Phân bổ tập hợp số nguyên

Bài toán: Cho 13 số nguyên bất kỳ. Chứng minh rằng tồn tại hai số có phần dư giống nhau khi chia cho 12.

Phân tích: Phần dư khi chia cho 12 thuộc tập {0, 1, ..., 11} ⇒ 12 giá trị khả dĩ (12 hộp). Có 13 số nguyên ⇒ 13 đối tượng.

Áp dụng nguyên lý Dirichlet: $\lceil \frac{13}{12} \rceil = 2$ ➔ Tồn tại ít nhất hai số cùng phần dư.

◈ Ví dụ 4 – Bắt tay trong một nhóm

Bài toán: Có 10 người trong một buổi gặp mặt. Mỗi người bắt tay với một số người khác trong nhóm. Chứng minh rằng tồn tại ít nhất hai người bắt tay với cùng số lượng người.

Phân tích: Mỗi người có thể bắt tay từ 0 đến 9 người, nhưng trong thực tế, nếu một người bắt tay với 9 người thì không thể có người nào bắt tay với 0 người (vì họ đã bắt tay nhau).

⇒ Số lượng khả dĩ thực tế của số cái bắt tay là từ 1 đến 8 ⇒ 8 giá trị (hộp).
Có 10 người (đối tượng) ⇒ Áp dụng Dirichlet ⇒ phải có **ít nhất 2 người bắt tay với cùng số người**.

Những ví dụ trên cho thấy **sức mạnh của nguyên lý Dirichlet** không nằm ở khả năng chỉ rõ cụ thể, mà ở việc **chứng minh sự tồn tại** — một đặc trưng nổi bật trong các bài toán tổ hợp, thuật toán, và phân tích logic trong ngành CNTT.

1.5 Ứng dụng thực tế

Nguyên lý Dirichlet không chỉ là một định lý trừu tượng trong toán học rời rạc mà còn là công cụ tư duy cực kỳ hiệu quả trong việc phân tích, phát hiện, và chứng minh tính tất yếu trong nhiều hệ thống thực tế, đặc biệt trong lĩnh vực công nghệ thông tin.

Dưới đây là một số ứng dụng tiêu biểu:

◈ 1. Phân tích thuật toán và dữ liệu

Trong thiết kế thuật toán, nguyên lý Dirichlet được sử dụng để **chứng minh va chạm hoặc trùng lặp** là không thể tránh khỏi. Điển hình như trong các bài toán về:

- **Hashing (băm dữ liệu):** Khi ánh xạ $n + 1$ phần tử vào n ô nhớ → chắc chắn có va chạm. Điều này lý giải vì sao các hàm băm luôn cần xử lý xung đột (collision handling).
- **Bộ nhớ đệm (caching):** Với số lượng truy cập vượt quá số khối bộ nhớ → sẽ có ít nhất một khối bị ghi đè.

◈ 2. An toàn thông tin và mật mã học

- **Mã hóa đối xứng / khóa rút gọn:** Nếu không gian khóa nhỏ hơn số lượng thông điệp có thể, thì tồn tại **ít nhất hai thông điệp sử dụng cùng khóa**, dẫn đến rủi ro bị giải mã sai.
- **Phát hiện tấn công lặp:** Trong một dãy dài các thao tác mạng, nếu có nhiều hơn nn truy vấn vào nn cổng/kênh, thì chắc chắn có ít nhất một cổng bị truy cập lặp.

◈ 3. Mạng máy tính và truyền thông

- **Quản lý IP động (DHCP):** Khi số lượng thiết bị vượt quá số lượng địa chỉ cấp phát, sẽ có địa chỉ IP bị trùng.
- **Giao thức chia sẻ kênh (TDMA, CSMA):** Giới hạn số khe thời gian → nếu nhiều thiết bị truyền đồng thời, sẽ có xung đột truyền tin.

◈ 4. Cơ sở dữ liệu và hệ thống phân tán

- **Phân mảnh dữ liệu:** Khi lưu trữ nhiều bản ghi hơn số phân vùng hoặc node, sẽ có node chứa nhiều bản ghi hơn các node khác.
- **Tính toàn vẹn dữ liệu:** Kiểm tra tính duy nhất của khóa chính (primary key) có thể quy về nguyên lý Dirichlet để phát hiện xung đột khi nhập liệu.

◈ 5. Phân tích rủi ro và lập lịch

- **Lập lịch thi / phòng học:** Nếu có nhiều lớp hơn số phòng → sẽ có ít nhất một phòng phải tổ chức ≥ 2 lớp → cần sắp xếp ca thi.
- **Phân tích lỗi hệ thống:** Nếu có nhiều yêu cầu truy cập tài nguyên hơn số tài nguyên sẵn có → hệ thống phải xử lý cạnh tranh tài nguyên.

Nguyên lý Dirichlet giúp **hiểu rõ giới hạn của các hệ thống rời rạc**, từ đó thiết kế giải pháp hợp lý hơn thay vì chỉ chạy theo tối ưu hóa cục bộ. Việc nắm vững nguyên lý này không chỉ giúp sinh viên giải các bài toán học thuật, mà còn phát triển **năng lực phân tích hệ thống – một kỹ năng then chốt trong ngành công nghệ thông tin**.

PHẦN 2: TRẮC NGHIỆM

✅ **Câu 1:**

Trong một lớp có 30 sinh viên, chứng minh rằng có ít nhất 3 người sinh cùng tháng.

A. Không chắc chắn

B. Chỉ có thể là 2 người
C. Chắc chắn có ít nhất 3 người sinh cùng tháng
D. Không đủ dữ kiện

✅ **Câu 2:**

Có 11 đôi tất được cất vào 10 ngăn kéo. Theo nguyên lý Dirichlet:
A. Tất cả ngăn đều có đúng 1 đôi
B. Một ngăn có ít nhất 2 đôi
C. Có ngăn không có đôi nào
D. Không có ngăn nào trùng

✅ **Câu 3:**

Có bao nhiêu người cần để đảm bảo có ít nhất 2 người cùng ngày sinh (không tính năm nhuận)?
A. 365
B. 366
C. 367
D. 368

✅ **Câu 4:**

Có 101 số tự nhiên từ 1 đến 200. Chứng minh tồn tại 2 số có hiệu bằng nhau.
A. Sai, vì khoảng cách lớn
B. Đúng, vì số lượng số > 100 hiệu khả dĩ
C. Không chắc chắn
D. Chỉ xảy ra khi có số lặp

✅ **Câu 5:**

Trong 13 số nguyên, tồn tại hai số có cùng phần dư khi chia cho 12.
A. Không chắc chắn
B. Chỉ đúng nếu có số lặp
C. Sai vì 13 không chia hết cho 12
D. Đúng theo nguyên lý Dirichlet

✅ **Câu 6:**

Một router có thể cấp phát 100 địa chỉ IP. Có 101 thiết bị yêu cầu IP. Kết luận nào sau đây là đúng?
A. Có thiết bị không có IP
B. Tồn tại trùng địa chỉ IP
C. Cả A và B đúng
D. Không có gì xảy ra

✅ **Câu 7:**

Trong 29 sự kiện được xếp vào 4 tuần, mỗi tuần 7 ngày, khẳng định nào là đúng?
A. Có ngày trống
B. Mỗi ngày có tối đa 1 sự kiện
C. Có ít nhất 1 ngày ≥ 2 sự kiện
D. Không thể xác định

✅ **Câu 8:**

Trong nhóm 10 người, mỗi người bắt tay một số người khác. Chứng minh tồn tại ít nhất hai người có số lần bắt tay bằng nhau.
A. Luôn đúng
B. Sai khi tất cả bắt tay khác nhau
C. Không đủ dữ kiện
D. Đúng chỉ khi số người chẵn

✅ **Câu 9:**

Cho tập gồm 65 dãy nhị phân độ dài 6. Tồn tại ít nhất hai dãy giống nhau?
A. Sai vì có 64 tổ hợp
B. Đúng vì số dãy vượt số tổ hợp
C. Đúng nếu có trùng
D. Không thể xảy ra

✅ **Câu 10:**

Một hàm băm ánh xạ từ 5000 chuỗi về 4096 giá trị. Kết luận?
A. Không thể xảy ra trùng
B. Phải có ít nhất một va chạm
C. Có thể xảy ra trùng nếu băm không đều
D. Không xác định

✅ **Câu 11:**

Trong 21 quả bóng được phân vào 6 hộp. Số quả tối thiểu có thể nằm trong một hộp là:

A. 4
B. 3
C. 5
D. 6

✅ **Câu 12:**

Có 9 người chọn số từ 1 đến 7. Chắc chắn có ít nhất:

A. 2 người trùng số
B. 3 người trùng số
C. Không người nào trùng số
D. Không xác định được

✅ **Câu 13:**

Trong một bảng tính có 27 cột. Có 200 giá trị được nhập ngẫu nhiên. Tối thiểu một cột có bao nhiêu giá trị?

A. 7
B. 8
C. 6
D. 9

✅ **Câu 14:**

Có 11 ứng viên nộp hồ sơ vào 10 vị trí. Khẳng định nào là chắc chắn đúng?

A. Có vị trí bị bỏ trống
B. Có ứng viên không được nhận
C. Có vị trí nhận ≥ 2 hồ sơ
D. Tất cả đều đúng

✅ **Câu 15:**

Tập gồm 101 số nguyên bất kỳ luôn tồn tại:

A. 2 số có hiệu bằng 1
B. 2 số chia hết cho nhau
C. 2 số có cùng phần dư chia 100
D. Không khẳng định được

PHẦN 3: BÀI TẬP TỰ LUẬN

🔰 Bài 1. Sinh viên và tháng sinh

Lớp học có 30 sinh viên. Hỏi có thể khẳng định chắc chắn rằng có ít nhất 3 sinh viên sinh cùng một tháng trong năm hay không?

Gợi ý:

- Có 12 tháng trong năm → 12 "hộp"
- 30 sinh viên → 30 "đối tượng"
- Dùng nguyên lý Dirichlet tổng quát: $\lceil \frac{m}{n} \rceil$

Nhận xét:
Bài toán quen thuộc giúp sinh viên nhận thức rằng "chỉ cần đủ lớn thì sự trùng lặp là tất yếu" – tư tưởng cốt lõi của nguyên lý Dirichlet.

🔰 Bài 2. Phân chia vật vào hộp

Có 21 vật phẩm cần phân vào 6 hộp. Chứng minh rằng có ít nhất một hộp chứa từ 4 vật trở lên.

Gợi ý:

- Xác định số hộp và số vật
- Áp dụng công thức Dirichlet tổng quát để tìm số lượng tối thiểu trong một hộp

Nhận xét:
Bài toán đơn giản nhưng giúp sinh viên hình dung trực quan vai trò của nguyên lý trong bài toán phân phối.

🔰 Bài 3. Hai số có hiệu bằng nhau

Chọn 51 số nguyên từ đoạn 1 đến 100. Chứng minh rằng tồn tại hai số có hiệu bằng nhau.

Gợi ý:

- Tổng số hiệu khác nhau giữa 2 số trong đoạn 1–100 là bao nhiêu?
- So sánh với số lượng cặp số có thể tạo ra

Nhận xét:
Một ví dụ ngụy biện trực tiếp nhưng tinh tế: không dùng công thức Dirichlet rõ ràng, mà vận dụng tư duy đếm số quan hệ.

⚙️ Bài 4. Chia dư cho 12

Chọn 13 số nguyên bất kỳ. Chứng minh rằng có ít nhất 2 số có cùng phần dư khi chia cho 12.

Gợi ý:

- Khi chia cho 12, ta thu được bao nhiêu giá trị phần dư?
- So sánh với số lượng số đang xét.

Nhận xét:
Bài toán thực tiễn liên quan đến số học modular – tiền đề cho lập trình bảo mật, kiểm tra xung đột trong hash.

⚙️ Bài 5. Lịch họp trong 4 tuần

Có 29 cuộc họp được lên lịch trong vòng 4 tuần (mỗi tuần 7 ngày). Chứng minh rằng có ít nhất một ngày có ≥ 2 cuộc họp.

Gợi ý:

- Tổng số ngày là bao nhiêu?
- Nếu mỗi ngày chỉ chứa tối đa 1 cuộc họp thì cần bao nhiêu ngày?

Nhận xét:
Bài toán giúp sinh viên tiếp cận nguyên lý Dirichlet trong bài toán lập lịch – kỹ năng thực tế trong phát triển phần mềm.

⚙️ Bài 6. Cấp phát địa chỉ IP

Một bộ định tuyến có thể cấp phát 100 địa chỉ IP. Nếu có 101 thiết bị yêu cầu, chứng minh rằng chắc chắn có ít nhất một địa chỉ IP bị trùng.

Gợi ý:

- IP là "hộp", thiết bị là "đối tượng"
- Số lượng thiết bị có vượt số IP không?

Nhận xét:
Ví dụ rõ ràng của nguyên lý Dirichlet trong hệ thống mạng – nền tảng cho sinh viên ngành CNTT.

🚀 CẤP ĐỘ 3 – NÂNG CAO

🧩 **Bài 7:** 10 người bắt tay trong nhóm. Mỗi người bắt tay với một số người khác.

Gợi ý:

- Suy nghĩ xem có thể có một người bắt tay với tất cả và một người không bắt tay ai không?
- Xác định các giá trị số lần bắt tay hợp lệ
- So sánh số người và số giá trị khả dĩ

Nhận xét: Một bài toán điển hình kết hợp Dirichlet với logic phản chứng.

🧩 **Bài 8:** Có 65 dãy nhị phân độ dài 6. Chứng minh có ít nhất 2 dãy giống nhau.

Gợi ý:

- Có bao nhiêu dãy nhị phân độ dài 6 khác nhau?
- Nếu tạo ra nhiều hơn số lượng dãy đó thì điều gì xảy ra?

Nhận xét: Một minh họa trực quan của hiện tượng "đụng độ không gian" (space collision), rất thường gặp trong lập trình, mã hóa.

🧩 **Bài 9:** Hàm băm ánh xạ 5000 chuỗi vào tập giá trị gồm 4096 phần tử.

Gợi ý:

- Có bao nhiêu đầu ra (hash value) khác nhau?
- Nếu số chuỗi đầu vào nhiều hơn số hash value, có trùng không?

Nhận xét: Đây là cách chứng minh không xây dựng (*non-constructive proof*) cho va chạm – kỹ thuật cực kỳ quan trọng trong lý thuyết thuật toán và bảo mật.

PHẦN 4: LÝ THUYẾT 2

B: CÁC CẤU HÌNH TỔ HỢP *(Combinatorial Configurations)*

🔍 2.1 Trực quan & Động lực

Trong nhiều bài toán thực tế và kỹ thuật, chúng ta thường không cần liệt kê tường minh các khả năng, mà chỉ cần biết có **bao nhiêu cách** để thực hiện một hành động nào đó.

Câu hỏi "Có bao nhiêu cách để...?" là cốt lõi của **kỹ thuật đếm tổ hợp**, và được biểu diễn dưới dạng các **cấu hình tổ hợp cơ bản** như: hoán vị, chỉnh hợp, tổ hợp.

Ví dụ thực tiễn trong ngành CNTT:

- Có bao nhiêu chuỗi ký tự độ dài 8 từ bảng chữ cái 26 chữ cái?
- Một mật khẩu có 4 ký tự chữ + 2 số, có bao nhiêu cách tạo ra?
- Một thuật toán cần duyệt qua mọi cách chọn 3 file từ thư mục chứa 100 file – cần duyệt bao nhiêu tổ hợp?

Từ góc nhìn trực quan:

- **Khi thứ tự quan trọng** → bài toán là **sắp xếp** (hoán vị, chỉnh hợp)
- **Khi thứ tự không quan trọng** → bài toán là **chọn** (tổ hợp)
- **Khi có lặp lại** → cấu hình trở nên phong phú hơn: tổ hợp có lặp, hoán vị lặp

Trong khoa học máy tính, tổ hợp là nền tảng cho:

- **Sinh cấu hình (configuration generation)**
- **Thiết kế thuật toán vét cạn / quay lui**
- **Phân tích độ phức tạp bài toán P vs NP**
- **Sinh dữ liệu kiểm thử (test case generation)**
- **Tối ưu hóa lựa chọn – bài toán ba lô (knapsack), bài toán tổ hợp tuyến tính, phân hoạch tập hợp**

Việc nắm vững các loại cấu hình tổ hợp và sử dụng đúng công thức không chỉ giúp sinh viên giải nhanh bài toán đếm, mà còn là nền tảng để hiểu sâu về các kỹ thuật như **quy hoạch động**, **truy hồi**, **lý thuyết xác suất rời rạc**, và **thuật toán sinh sinh (generative algorithms)** trong chuyên ngành CNTT.

📘 2.2 Các cấu hình cơ bản

Các cấu hình tổ hợp cơ bản bao gồm ba loại chính: **hoán vị**, **chỉnh hợp**, và **tổ hợp**. Mỗi loại tương ứng với một cách thức **chọn và sắp xếp phần tử** từ một tập hợp, tùy theo việc **thứ tự có quan trọng hay không** và **có chọn toàn bộ hay chỉ một phần**.

◈ 1. Hoán vị (Permutation)

Định nghĩa:
Hoán vị là số cách **sắp xếp toàn bộ** n phần tử **khác nhau** theo một thứ tự nhất định.

Ký hiệu:

$P(n) = n!$

Ví dụ:
Có bao nhiêu cách sắp xếp 4 học sinh vào 4 ghế?

$P(4) = 4! = 24 \text{ cách}$

Ứng dụng trong CNTT:

- Sinh tất cả hoán vị đầu vào cho kiểm thử hệ thống
- Duyệt toàn bộ thứ tự thực thi tác vụ
- Sắp xếp trạng thái trong thuật toán backtracking

◈ 2. Chỉnh hợp (Arrangement)

Định nghĩa:
Chỉnh hợp là số cách **chọn và sắp xếp r phần tử** từ n phần tử, **có phân biệt thứ tự**.

Ký hiệu:

$$A(n\,,\,r) = \frac{n!}{(n-r)!}$$

Ví dụ:
Chọn 3 người từ 5 người để làm tổ trưởng, thư ký, thủ quỹ (phân biệt vai trò).

$$A(5\,,\,3) = \frac{5!}{(5-3)!} = \frac{120}{2} = 60$$

Ứng dụng trong CNTT:

- Lập lịch CPU theo thứ tự ưu tiên
- Sinh không gian tìm kiếm có trọng số (ordering-sensitive search)
- Tạo mã đăng nhập theo thứ tự ký tự

◈ 3. Tổ hợp (Combination)

Định nghĩa:
Tổ hợp là số cách **chọn r phần tử** từ n*n* phần tử **khác nhau**, **không phân biệt thứ tự**.

Ký hiệu:

$$C(n\,,\,r) = \binom{n}{r} = \frac{n!}{r!(n-r)!}$$

Ví dụ:
Chọn 3 sinh viên từ 10 người để lập nhóm thuyết trình (không phân biệt vai trò):

$$C(10\ ,\ 3) = \frac{10!}{3!7!} = 120$$

Ứng dụng trong CNTT:

- Sinh tất cả các tập con kích thước cố định
- Xác định các tập tham số trong kiểm thử tổ hợp
- Tối ưu hóa truy vấn CSDL (các lựa chọn ràng buộc)

✅ **So sánh tổng quát:**

Tình huống	**Tên gọi**	**Công thức**
Sắp xếp toàn bộ	Hoán vị	n!
Chọn và sắp xếp r phần tử	Chỉnh hợp	$\frac{n!}{(n-r)!}$
Chọn r phần tử (không sắp xếp)	Tổ hợp	$\frac{n!}{r!(n-r)!}$

Việc xác định đúng **loại cấu hình** là bước đầu tiên và quan trọng nhất khi tiếp cận các bài toán tổ hợp. Trong nhiều tình huống thực tế, **việc chọn đúng công thức giúp giải bài nhanh, chính xác và tiết kiệm tài nguyên tính toán** – đặc biệt quan trọng trong môi trường xử lý dữ liệu lớn, lập trình giải thuật, hoặc phân tích tổ hợp trạng thái trong hệ thống.

🔁 2.3 Các cấu hình có lặp

Trong thực tế, nhiều bài toán yêu cầu **chọn hoặc sắp xếp phần tử có thể trùng nhau** – ví dụ: chọn nhiều viên kẹo cùng loại, tạo mật khẩu có ký tự lặp lại, hoặc sắp xếp các ký tự giống nhau trong một từ. Các tình huống này dẫn đến các **cấu hình tổ hợp có lặp**, gồm hai loại chính:

💠 1. Hoán vị lặp (Permutations with repetition)

Định nghĩa:
Là số cách sắp xếp nn phần tử, trong đó có **các phần tử trùng nhau**, tức là **không phân biệt toàn bộ phần tử**.

Công thức:

$$P = \frac{n!}{k_1! \cdot k_2! \cdot \ldots \cdot k_m!}$$

Trong đó:

- n: tổng số phần tử

  $k_1, k_2, ..., k_m$: số lần xuất hiện của mỗi phần tử giống nhau

Ví dụ:
Từ “SUCCESS” có bao nhiêu cách sắp xếp các chữ cái?

- Tổng số chữ: 7
- Chữ S lặp 3 lần, C lặp 2 lần, còn lại U và E xuất hiện 1 lần

$$P = \frac{7!}{3! \cdot 2! \cdot 1! \cdot 1!} = \frac{5040}{12} = 420$$

Ứng dụng trong CNTT:

- Đếm số cách xếp các ký tự trong chuỗi chứa ký tự lặp
- Sinh tổ hợp đầu vào có thứ tự không phân biệt
- Kiểm thử thuật toán xử lý xâu có dữ liệu trùng

🔷 2. Tổ hợp có lặp (Combinations with repetition)

Định nghĩa:
Là số cách chọn r phần tử từ n loại phần tử **cho phép lặp**, **không phân biệt thứ tự**.

Công thức:

$$C'(n\ ,\ r) = \binom{n + r - 1}{r}$$

Ví dụ:
Có 3 loại bánh: A, B, C. Chọn ra 5 chiếc bánh, không phân biệt thứ tự, có thể chọn nhiều bánh cùng loại.

$$C'(3\ ,\ 5) = \binom{3 + 5 - 1}{5} = \binom{7}{5} = 21$$

Ứng dụng trong CNTT:

- Sinh tập giá trị cho biến có thể chọn lặp (ví dụ: giá trị màu, ký hiệu)
- Phân phối tài nguyên lặp lại (bài toán phân phối không phân biệt thứ tự)
- Tạo tổ hợp dữ liệu từ nhóm tùy chọn có số lượng không giới hạn

🧠 Lưu ý phân biệt dễ nhầm

Câu hỏi	Loại cấu hình	Lặp	Thứ tự	Công thức
Sắp xếp n phần tử khác nhau	Hoán vị	✗	✔	n!n!
Sắp xếp n phần tử có trùng	Hoán vị lặp	✔	✔	$\frac{n!}{k_1!\cdot\ldots}$
Chọn r từ n phần tử (không lặp)	Tổ hợp	✗	✗	$\binom{n}{r}$
Chọn r từ n loại, có thể lặp	Tổ hợp có lặp	✔	✗	$\binom{n+r-1}{r}$

Các cấu hình có lặp là cầu nối giữa tổ hợp thuần túy và các bài toán **mô hình hóa dữ liệu có tính lặp**, vốn rất phổ biến trong lập trình, hệ thống cơ sở dữ liệu, và tối ưu hóa. Việc nhận diện đúng cấu trúc bài toán sẽ giúp sinh viên áp dụng chính xác công thức và phát triển tư duy thuật toán bền vững.

📊 2.4 Tóm tắt phân loại & sơ đồ tổng hợp

Trong thực hành tổ hợp, điểm khó không nằm ở công thức, mà ở **việc xác định đúng cấu hình tổ hợp cần sử dụng**. Các bài toán tổ hợp thường rơi vào một trong 4 tình huống cơ bản sau:

◈ A. Bảng phân loại cấu hình tổ hợp

Tình huống bài toán	Phân biệt thứ tự	Có lặp?	Tên gọi	Công thức
Sắp xếp **toàn bộ n phần tử phân biệt**	✔	✗	Hoán vị	$P(n) = n!$
Chọn r phần tử **và sắp xếp**, không lặp	✔	✗	Chỉnh hợp	$A(n\ ,\ r) = \frac{n!}{(n\ -\ r)!}$
Chọn r phần tử **không sắp xếp**, không lặp	✗	✗	Tổ hợp	$C(n\ ,\ r) = \binom{n}{r}$
Chọn r phần tử **không sắp xếp, có lặp**	✗	✔	Tổ hợp có lặp	$\binom{n+r-1}{r}$
Sắp xếp n phần tử **có trùng lặp**	✔	✔	Hoán vị lặp	$\frac{n!}{k_1!\cdot\ldots}$

🔷 B. Sơ đồ tổng hợp trực quan

```
                        [ Bài toán đếm tổ hợp ]
                                   |
                 +-----------------+-----------------+
                 |                                   |
          Có phân biệt thứ tự                Không phân biệt thứ tự
                 |                                   |
          +------+-------+                +----------+-----------+
          |              |                |                      |
    Không cho lặp     Có lặp        Không cho lặp              Có lặp
          |              |                |                      |
    Chỉnh hợp         Lũy thừa      Tổ hợp                 Tổ hợp có lặp
 A(n, r) = n!/(n - r)!   n^r     C(n, r) = n!/(r!(n - r)!)  C’(n, r) = C(n + r - 1, r)
```

🧠 C. Quy tắc nhận diện nhanh (4 bước)

Câu hỏi	**Nếu trả lời “Có” thì...**
Thứ tự có quan trọng không?	✅ → Dùng **hoán vị / chỉnh hợp**
Có chọn toàn bộ hay một phần?	Toàn bộ → **hoán vị**, một phần → **chỉnh hợp**
Có cho phép lặp không?	✅ → Dùng lũy thừa hoặc tổ hợp có lặp
Các phần tử có trùng không?	✅ → **Hoán vị lặp**

☑️ D. Ví dụ ứng dụng tổng hợp

Bài toán	**Nhận diện cấu hình**	**Công thức áp dụng**
Sắp xếp 5 người vào 5 chỗ ngồi khác nhau	Hoán vị	$5! = 120$
Chọn 3 người làm chức danh khác nhau từ 6 ứng viên	Chỉnh hợp	$A(6\ ,\ 3) = 120$
Chọn 4 món ăn từ 10 món, không phân biệt thứ tự	Tổ hợp	$C(10\ ,\ 4) = 210$
Chọn 3 viên bi từ 5 loại, cho phép chọn trùng loại	Tổ hợp có lặp	$\binom{5+3-1}{3} = 35$
Sắp xếp từ “LEVEL” gồm 5 chữ cái (có lặp)	Hoán vị lặp	$\frac{5!}{2!2!1!} = 30$

📌 Kết luận

Việc **thành thạo phân loại cấu hình tổ hợp** giúp sinh viên:

- Giải bài nhanh, chính xác, không nhầm lẫn
- Tiết kiệm thời gian khi xử lý không gian tìm kiếm lớn
- Ứng dụng linh hoạt trong thuật toán, kiểm thử phần mềm, mật mã học và dữ liệu lớn

PHẦN 5: TRẮC NGHIỆM 2

✅ **Câu 1 (Dễ):**

Sắp xếp 4 học sinh vào 4 chỗ ngồi khác nhau, có bao nhiêu cách?
A. 16
B. 24
C. 12
D. 10

✅ **Câu 2:**

Chọn 2 người từ 5 người để làm nhóm, không phân biệt thứ tự. Có bao nhiêu cách chọn?
A. 10
B. 20
C. 25
D. 5

✅ **Câu 3:**

Tạo mã gồm 3 chữ số, cho phép trùng số. Có bao nhiêu mã?
A. 100
B. 900
C. 1000
D. 729

✅ **Câu 4:**

Tạo mã gồm 3 chữ cái khác nhau từ bảng chữ cái tiếng Anh. Có bao nhiêu mã?
A. 26^3
B. $C(26, 3)$
C. $A(26, 3)$
D. $3! \cdot 26$

✅ **Câu 5:**

Chọn 3 viên kẹo từ 5 loại, **cho phép chọn trùng loại**, không quan trọng thứ tự. Số cách?

A. $C(5, 3)$

B. 5^3

C. $\binom{7}{3}$

D. $A(5, 3)$

✅ **Câu 6:**

Từ "LEVEL", có bao nhiêu cách sắp xếp các chữ cái?

A. 120

B. 60

C. 30

D. 10

✅ **Câu 7:**

Chọn 3 người từ 8 người để lập nhóm. Thứ tự **không quan trọng**. Cấu hình nào đúng?

A. Hoán vị

B. Chỉnh hợp

C. Tổ hợp

D. Tổ hợp có lặp

✅ **Câu 8:**

Chọn 3 chữ số khác nhau từ 0–9 để tạo mã. Bao nhiêu mã nếu **có xét thứ tự và không dùng lại chữ số**?

A. 720

B. 1000

C. 120

D. 504

✅ **Câu 9:**

Có bao nhiêu cách chọn 4 món ăn từ 10 món, không trùng, không phân biệt thứ tự?

A. $A(10, 4)$

B. 10^4

C. $C(10\ ,\ 4)$
D. 4!

✅ **Câu 10:**

Có bao nhiêu xâu nhị phân độ dài 6?
A. 64
B. 128
C. 36
D. 32

✅ **Câu 11:**

Từ “SUCCESS”, có bao nhiêu cách sắp xếp các chữ cái?
A. 840
B. 5040
C. 420
D. 360

✅ **Câu 12:**

Tạo mật khẩu gồm 3 chữ và 2 số (chữ khác nhau, số có thể lặp). Số cách?
A. $C(26\ ,\ 3) \cdot 10^2$
B. $A(26\ ,\ 3) \cdot 10^2$
C. $26^3 \cdot 10^2$
D. $A(26\ ,\ 3) \cdot A(10\ ,\ 2)$

✅ **Câu 13:**

Có bao nhiêu cách xếp 3 quả bóng vào 5 ngăn tủ (1 quả mỗi ngăn), không giới hạn số bóng trong mỗi ngăn?
A. 5^3
B. $A(5\ ,\ 3)$
C. $C(5\ ,\ 3)$
D. $\binom{7}{3}$

✅ **Câu 14:**

Có 5 loại bánh, chọn 7 chiếc (có thể chọn trùng loại). Bao nhiêu cách chọn?
A. $\binom{5+7-1}{7}$

B. $C(5, 7)$
C. $A(5, 7)$
D. 5^7

✅ **Câu 15:**

Có bao nhiêu số tự nhiên gồm 4 chữ số khác nhau từ 0–9, chữ số đầu tiên **không thể là 0**?
A. $9 \cdot A(9, 3)$
B. $A(10, 4)$
C. $9 \cdot C(9, 3)$
D. $A(9, 4)$

PHẦN 6: BÀI TẬP LUYỆN TẬP 2

🔰 Bài 1 (Dễ): Sắp xếp ghế ngồi

Đề: Có 4 sinh viên và 4 ghế ngồi thẳng hàng. Hỏi có bao nhiêu cách sắp xếp chỗ ngồi cho các sinh viên?

Gợi ý: Đây là bài toán **hoán vị toàn bộ** 4 phần tử.

Nhận xét: Bài tập cơ bản để làm quen với khái niệm hoán vị.

🔰 Bài 2: Chọn nhóm thuyết trình

Đề: Từ 6 sinh viên, chọn ra 3 người để lập nhóm thuyết trình. Không phân biệt vai trò.

Gợi ý: Vì không phân biệt vai trò ⇒ bài toán tổ hợp không lặp.

Nhận xét: Câu hỏi phổ biến kiểm tra khả năng nhận biết khi nào "thứ tự không quan trọng".

🔰 Bài 3: Tạo mã số có lặp

Đề: Một mã số gồm 3 chữ số từ 0–9. Cho phép trùng lặp. Có bao nhiêu mã số?

Gợi ý: Mỗi chữ số có 10 lựa chọn, có lặp ⇒ dùng lũy thừa.

Nhận xét: Giới thiệu về đếm với lặp lại – bước đệm để học tổ hợp có lặp.

🔰 Bài 4: Chọn sách

Đề: Chọn 2 cuốn sách từ kệ có 10 cuốn khác nhau. Thứ tự không quan trọng.

Gợi ý: Dùng tổ hợp, không xét thứ tự.

Nhận xét: Một ví dụ tổ hợp đơn giản, thường gặp trong bài toán chọn.

⚙️ Bài 5: Xếp người vào ban cán sự

Đề: Chọn 3 người từ 8 sinh viên để phân công làm lớp trưởng, lớp phó, bí thư (mỗi vai trò khác nhau). Hỏi có bao nhiêu cách?

Gợi ý: Bài toán có thứ tự ⇒ chỉnh hợp.

Nhận xét: Bài toán giúp phân biệt rõ tổ hợp và chỉnh hợp.

⚙️ Bài 6: Chọn kẹo có thể trùng

Đề: Có 4 loại kẹo khác nhau. Chọn 6 viên kẹo bất kỳ, cho phép chọn trùng loại. Hỏi có bao nhiêu cách chọn?

Gợi ý: Chọn có lặp, không phân biệt thứ tự → tổ hợp có lặp.

Nhận xét: Giúp sinh viên hiểu cấu hình tổ hợp có lặp – thường khó nhận diện.

⚙️ Bài 7: Mã hóa từ chữ cái

Đề: Tạo tất cả chuỗi ký tự độ dài 3 gồm các chữ cái in hoa tiếng Anh (A–Z), không lặp chữ.

Gợi ý: Có 26 ký tự, không lặp, có thứ tự ⇒ chỉnh hợp.

Nhận xét: Ứng dụng thực tế trong sinh mã, mã hóa, kiểm thử tổ hợp ký tự.

🚀 Bài 8: Sắp xếp chữ trong từ “BANANA”

Đề: Có bao nhiêu cách sắp xếp các chữ cái trong từ “BANANA”?

Gợi ý: Hoán vị lặp: từ có 6 chữ, với A(3 lần), N(2 lần), B(1 lần)

Nhận xét: Kiểu bài dễ nhầm nếu không nhận diện hoán vị có trùng.

🚀 Bài 9: Sinh chuỗi nhị phân

Đề: Có bao nhiêu chuỗi nhị phân độ dài 8 có đúng 3 số 1?

Gợi ý: Chọn 3 vị trí đặt số 1 trong 8 vị trí → tổ hợp.

Nhận xét: Bài toán thực tế trong sinh test case, mã nhị phân có trọng số cố định.

🚀 Bài 10 (Khó): Số tự nhiên 4 chữ số khác nhau, không bắt đầu bằng 0

Đề: Có bao nhiêu số tự nhiên gồm 4 chữ số khác nhau từ 0–9, trong đó chữ số đầu tiên **không phải 0**?

Gợi ý:

- Chữ đầu: 1–9 → 9 lựa chọn
- Các chữ tiếp theo: chọn từ 9 số còn lại (vì không lặp)
  → Dùng chỉnh hợp kết hợp điều kiện

Nhận xét: Bài toán nâng cao, tích hợp nhiều kỹ năng tổ hợp + điều kiện ràng buộc.

CHEAT SHEET TỔ HỢP (COMBINATORICS) - PHIÊN BẢN A4 DÀNH CHO SINH VIÊN CNTT

📊 TÓM Tắt PHÂN LOẠI CẤU HÌNH TỔ HỢP

Tình huống bài toán	Phân biệt thứ tự	Có lặp?	Tên gọ i	Công thức
Sắp xếp toàn bộ n phần tử khác nhau	✅	❌	Hoán vị	$P(n) = n!$
Chọn r và sắp xếp, không lặp	✅	❌	Chỉnh hợp	$A(n, r) = \frac{n!}{(n-r)!}$
Chọn r không sắp xếp, không lặp	❌	❌	Tổ hợp	$C(n, r) = \frac{n!}{r!(n-r)!}$
Chọn r, không sắp xếp, **có lặp**	❌	✅	Tổ hợp có lặp	$C'(n, r) = \binom{n+r-1}{r}$
Sắp xếp n phần tử có trùng lặp	✅	✅	Hoán vị lặp	$\frac{n!}{k_1! \cdot \ldots \cdot k_m!}$

🔄 Sơ đồ tổng hợp trực quan

```
                     [ Bài toán đếm tổ hợp ]
                                |
              +-----------------+-------------------+
              |                                     |
        Có phân biệt thứ tự                Không phân biệt thứ tự
              |                                     |
        +------+-----+                    +-------+--------+
        |            |                    |              |
   Không lặp     Có lặp              Không lặp        Có lặp
        |            |                    |              |
   Chỉnh hợp     Lũy thừa                Tổ hợp     Tổ hợp có lặp
  A(n, r) = n!/(n-r)!    n^r        C(n, r) = n!/(r!(n-r)!)   C'(n, r) = C(n+r-1, r)
```

✴️ Quy tắc nhận diện nhanh (4 bước)

1. **Thứ tự có quan trọng không?**
    - Có ✅ → Hoán vị / Chỉnh hợp
2. **Chọn toàn bộ hay một phần?**
    - Toàn bộ → Hoán vị, Một phần → Chỉnh hợp
3. **Cho lặp không?**
    - Có ✅ → Tổ hợp có lặp / Lũy thừa
4. **Các phần tử có trùng lặp?**
    - Có ✅ → Hoán vị lặp

📐 Ví dụ áp dụng

Bài toán	Nhận diện cấu hình	Công thức áp dụng
Sắp xếp 5 người vào 5 chỗ ngồi khác nhau	Hoán vị	$5! = 120$
Chọn 3 người làm chức danh từ 6 ứng viên	Chỉnh hợp	$A(6\ ,\ 3) = 120$
Chọn 4 món ăn từ 10 món, không phân biệt thứ tự	Tổ hợp	$C(10\ ,\ 4) = 210$
Chọn 3 viên bi từ 5 loại, có thể trùng loại	Tổ hợp có lặp	$\binom{7}{3} = 35$
Sắp xếp từ “LEVEL” gồm 5 chữ (có lặp)	Hoán vị lặp	$\frac{5!}{2!2!1!} = 30$

Ghi nhớ: Chìa khoá để làm đúng là: *Xác định đúng yêu cầu của bài: có thứ tự không? có lặp không? có sắp xếp không?*

📓 CHEAT SHEET TỔ HỢP (COMBINATORICS)

Phiên bản A4 – Dành cho sinh viên CNTT

📊 Tóm tắt phân loại cấu hình tổ hợp

Tình huống bài toán	Phân biệt thứ tự	Có lặp?	Tên gọi	Công thức
Sắp xếp toàn bộ n phần tử khác nhau	✅	❌	**Hoán vị**	$P(n) = n!$
Chọn r phần tử và sắp xếp (không lặp)	✅	❌	**Chỉnh hợp**	$A(n, r) = \frac{n!}{(n-r)!}$
Chọn r phần tử (không sắp xếp, không lặp)	❌	❌	**Tổ hợp**	$C(n, r) = \frac{n!}{r!(n-r)!}$
Chọn r phần tử (không sắp xếp, có lặp)	❌	✅	**Tổ hợp có lặp**	$C'(n, r) = \binom{n+r-1}{r}$
Sắp xếp n phần tử có trùng lặp	✅	✅	**Hoán vị lặp**	$\frac{n!}{k_1! \cdot k_2! \cdots k_m!}$

🔄 Sơ đồ tổng hợp trực quan

```
                      [ Bài toán đếm tổ hợp ]
                                |
             +------------------+--------------------+
             |                                       |
      Có phân biệt thứ tự                   Không phân biệt thứ tự
             |                                       |
      +------+-----+                         +-------+--------+
      |            |                         |                |
   Không lặp    Có lặp                   Không lặp         Có lặp
      |            |                         |                |
   Chỉnh hợp    Lũy thừa                    Tổ hợp      Tổ hợp có lặp
 A(n, r) = n!/(n-r)!   n^r          C(n, r) = n!/(r!(n-r)!)   C'(n, r) = C(n+r-1, r)
```

✅ Quy tắc nhận diện nhanh (4 bước)

1. **Thứ tự có quan trọng không?**
   - ✔️ Có → Hoán vị / Chỉnh hợp
   - ❌ Không → Tổ hợp / Tổ hợp có lặp

2. **Chọn toàn bộ hay một phần?**
    - Toàn bộ → Hoán vị
    - Một phần → Chỉnh hợp hoặc Tổ hợp
3. **Có cho phép lặp phần tử không?**
   ✅ Có → Tổ hợp có lặp / Lũy thừa
   ❌ Không → Hoán vị / Chỉnh hợp / Tổ hợp
4. **Có phần tử bị trùng không (dạng sẵn)?**
   ✅ Có → Hoán vị lặp (ví dụ: sắp xếp chữ "LEVEL")

📌 Ví dụ minh họa

Bài toán	**Nhận diện cấu hình**	**Công thức áp dụng**
Sắp xếp 5 người vào 5 chỗ ngồi khác nhau	Hoán vị	$5! = 120$
Chọn 3 người làm chức danh khác nhau từ 6 ứng viên	Chỉnh hợp	$\mathrm{A}(6\ ,\ 3) = 120$
Chọn 4 món ăn từ 10 món, không phân biệt thứ tự	Tổ hợp	$\mathrm{C}(10\ ,\ 4) = 210C$
Chọn 3 viên bi từ 5 loại, có thể trùng loại	Tổ hợp có lặp	$\binom{5+3-1}{3} = 35$
Sắp xếp các chữ trong từ "LEVEL" (có lặp)	Hoán vị lặp	$\frac{5!}{2!2!1!} = 30$

📌 Ghi nhớ quan trọng:

❗ Đừng cố nhớ công thức trước. Hãy **nhận diện đúng bài toán** bằng cách đặt 3 câu hỏi:

- Thứ tự có quan trọng không?
- Có phần tử trùng không?
- Có được chọn lặp lại không?