SAMSUNG 삼성SDS

SW 아카데미 알고리즘 특강

Contents

오늘의 원리

- 특정한 상황에서의 경우의 수를 계산함에 있어 필요한 조합론의 기본적인 계산법을 익히고 이를 활용하여 문제를 해결하는 방법을 배운다.
- 이항정리와 이항계수를 이해하고, 이를 조합의 성질을 활용하여 표현할 수 있다.

학습목표

- 조합론의 기본적인 계산방법과 성질을 공부하여, 특정 조건을 만족하는 경우의 수를 구하는 방법을 습득할 수 있다.
- 조합의 성질을 활용하여 주어진 문제를 부분문제로 나누는 방법을 습득할 수 있다.

연습문제

1번부터 10번까지 번호가 매겨진 10자루의 볼펜과 11번부터 15번까지 번호를 매긴 5자루의 연필이 있을 때, 아래 문제를 해결하시오.

1. 필기구 한 자루를 선택하는 경우의 수는?
2. 볼펜과 연필 각각 한 자루씩 총 두 자루를 선택하는 경우의 수는?
3. 필기구 두 자루를 선택하여 나열하는 경우의 수는?
4. 필기구 두 자루를 선택하는 경우의 수는?

볼펜, 연필 두 가지의 필기구가 있을 때, 아래 문제를 해결하시오.

5. 중복을 허용하여 필기구 네 자루를 선택해 나열하는 경우의 수는?
6. 중복을 허용하여 필기구 세 자루를 선택하는 경우의 수는?

연습문제

(1) 15자루의 필기구 중 1자루를 선택하는 방법은 15가지

(2) 10자루의 볼펜 중 1자루를 고르고, 5자루의 연필 중 1자루를 고르는 방법은 $10 \times 5 = 50$가지

(3) 첫번째 나열할 필기구를 15자루 중 1자루를 고르고,
두번째 나열할 필기구를 남은 14자루 중 1자루를 고르는 방법은 $15 \times 14 = 210$가지

(4) 필기구 두 자루를 선택하여 나열한 경우의 수에서
첫번재와 두번째 순서가 다른 경우의 수가 2번씩 존재하므로
필기구 두 자루를 선택하는 방법은 $15 \times 14 \div 2 = 105$가지

연습문제

(5) 첫 번째 나열할 필기구는 볼펜, 연필 두 자루를 모두 고를 수 있고,
중복을 허용하므로 두 번째, 세 번째, 네 번째 필기구도 두 자루를 모두 고를 수 있다.
따라서 $2 \times 2 \times 2 \times 2 = 16$가지

(6) 가능한 경우의 수는 아래와 같이 4가지이다.

볼펜3 연필0	볼펜	볼펜	볼펜	/
볼펜2 연필1	볼펜	볼펜	/	연필
볼펜1 연필2	볼펜	/	연필	연필
볼펜0 연필3	/	연필	연필	연필

이는 선택하는 필기구 3자루와 구분선 1개를 합친 4개의 영역에서 구분선의 위치를 선택하는 것과 동일한 문제이다.

구분선 1개의 위치를 선택하는 경우의 수는, 총 4개의 영역이 존재하므로 4가지이다.

순열과 조합

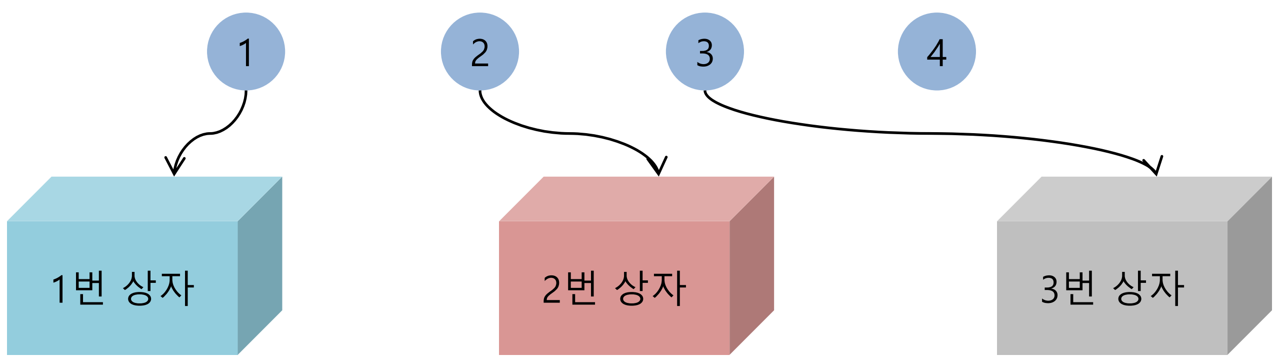
- **순열($_N P_K$) : N가지의 물건 중 K개의 물건을 순서 구분하여 고르는 경우의 수**

1부터 4까지의 숫자 중 3개의 숫자를 순서 구분하여 고르는 경우($_4 P_3$)

아래 그림처럼 세 개의 상자에 번호가 쓰여진 4개의 공을 담는 것과 같은 문제이다. 1번 상자에 담을 수 있는 숫자는 4개, 2번 상자에 담을 수 있는 숫자는 1번 상자에 담았던 숫자를 제외한 3개, 3번 상자에 담을 수 있는 숫자는 1,2번 상자에 담았던 숫자를 제외한 2개이다. 따라서, 총 경우의 수는 4×3×2 = 24가지이다.

다시 말해, N개의 물건 중 K개의 물건을 순서 구분하여 고르는 경우의 수는

$_N P_K = \underbrace{N \times (N-1) \times \cdots \times \big(N-(K-2)\big) \times \big(N-(K-1)\big)}_{K\text{개}} = \frac{N!}{(N-K)!}$으로 나타낼 수 있다.



순열과 조합

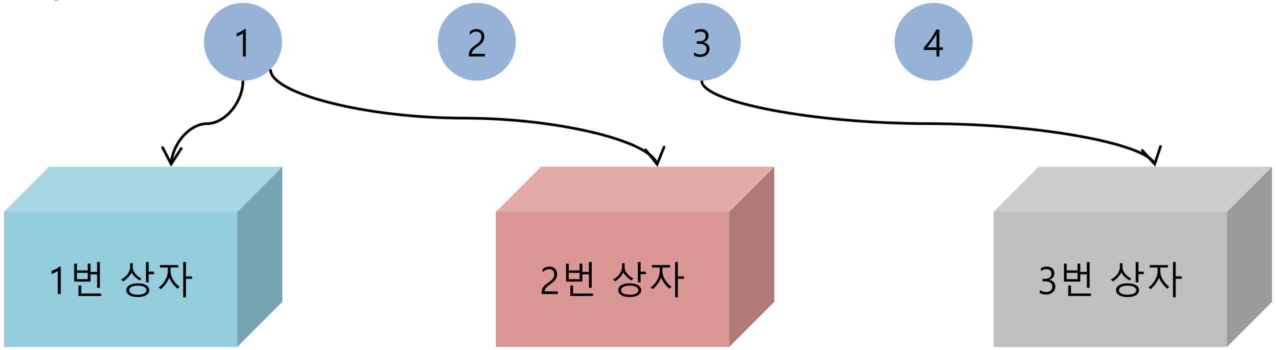
- **중복순열($_N\Pi_K$) : N가지의 물건 중 중복을 허용하여 K개의 물건을 순서 있게 고르는 경우의 수**

1부터 4까지의 숫자 중 중복을 허용하여 3개의 숫자를 순서 있게 고르는 경우($_4\Pi_3$)

1번 상자에 담을 수 있는 숫자는 4개, 2번 상자에 담을 수 있는 숫자는 1번 상자에 담았던 숫자도 담을 수 있어서 4개, 3번 상자에 담을 수 있는 숫자는 1,2번 상자에 담았던 숫자도 담을 수 있어서 4개이다. 따라서, 총 경우의 수는 4×4×4 = 64가지이다.

다시 말해, N개의 물건 중 중복을 허용하여 K개의 물건을 순서 있게 고르는 경우의 수는

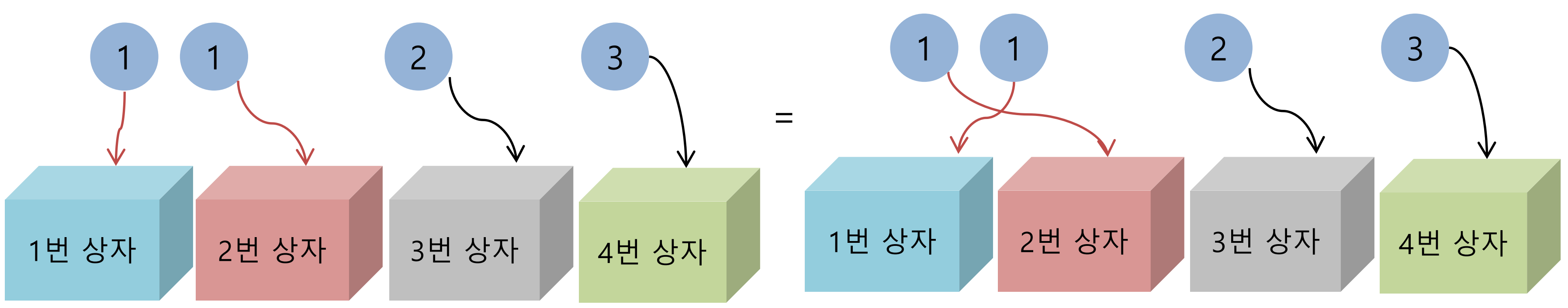
$_N\Pi_K = \underbrace{\mathrm{N} \times \mathrm{N} \times \cdots \times \mathrm{N}}_{K\text{개}} = N^K$으로 나타낼 수 있다.



순열과 조합

- **같은 것이 있는 순열(다중집합의 순열) : N가지의 물건 중 같은 물건이 각각 p, q, r 개 일 때, N개의 물건을 모두 택하여 순서 있게 고르는 경우의 수**

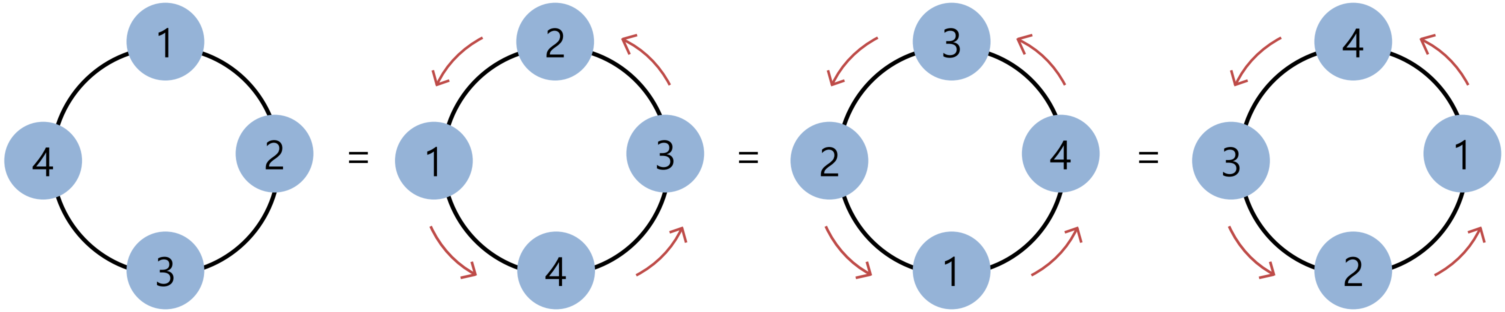
아래 그림처럼 네 개의 상자에 번호가 쓰여진 3개의 공을 담는 것과 같은 문제이다. 이 때 숫자 1이 쓰여진 공이 2개이다. 서로 다른 4개의 숫자(1,2,3,4) 를 상자에 담는 방법은 4! 이지만, 아래 그림과 같이 숫자 1이 1,2번 상자에 들어가는 2가지 경우가 같은 경우로 간주되므로 가능한 순열의 수는 $\frac{4!}{2!}$ 이다.
다시 말해, N가지의 물건 중 같은 물건이 각각 p, q, r개 일 때, N개의 물건을 모두 택하여 만든 순열의 수는 $\frac{N!}{p!q!r!}$ 으로 나타낼 수 있다.



순열과 조합

- **원순열 : N가지의 물건 중 K개의 물건을 원형으로 배열하는 경우의 수**

1부터 6까지의 숫자가 써있는 6개의 공 중에서 4개를 선택하여 원형으로 배열해보자. 원상의 위치가 모두 구별 가능하다고 가정하면 원소들을 선형으로 배열하는 방법의 수와 같으므로 ${}_6P_4$ 이다. 그런데 원형으로 배열하는 경우는 아래와 같이 회전에 의하여 같아지는 배열을 모두 동일하다고 간주한다. 따라서 중복되는 배열이 네 개씩 있으므로 총 경우의 수는 ${}_6P_4$ 을 4로 나눈 90가지이다.
다시 말해, 서로 다른 N가지의 물건 중 K개의 물건을 원형으로 배열하는 경우의 수는 $\frac{{}_NP_K}{K}$ 로 나타낼 수 있다.

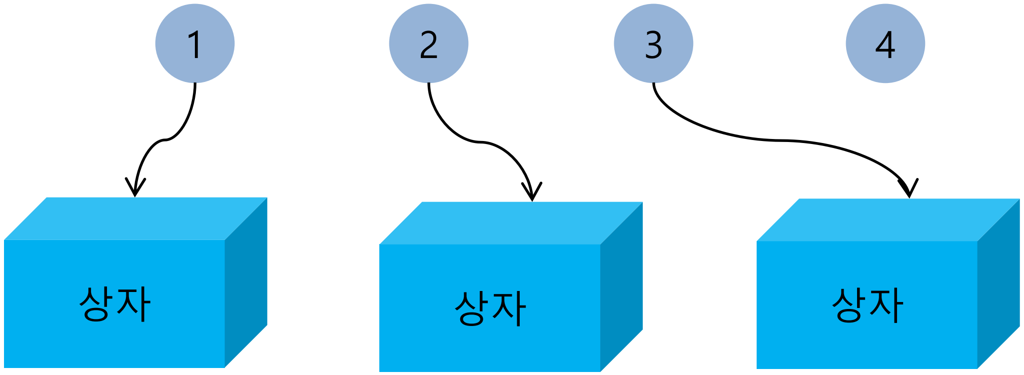


순열과 조합

- **조합($_NC_K$) : N가지의 물건 중 K개의 물건을 순서 구분 없이 고르는 경우의 수**

N가지의 물건 중 K개의 물건을 순서 구분하여 고르는 경우의 수는 $_NP_K(=\frac{N!}{(N-K)!})$이지만 K개의 물건을 순서 구분 없이 고르는 경우, K개의 물건을 고르는 경우의 수($K!$)가 모두 같은 경우로 간주되므로 $_NC_K$ = $_NP_K$ / K! 로 표현할 수 있다. (∴ $_NC_K=\frac{N!}{(N-K)!(K!)}$)

이를 통해 N가지의 물건 중 K개의 물건을 고르는 경우의 수와 N가지의 물건 중 N-K의 물건을 고르는 경우의 수는 같음을 알 수 있다. ($_NC_K={}_NC_{N-K}$)



순열과 조합

- **중복조합($_nH_k$) : N가지의 물건 중 중복을 허용하여 K개의 물건을 순서 구분 없이 고르는 경우의 수**

3개의 상자에 각각 1, 2 숫자가 쓰여진 2개의 공을 중복을 허용하여 담는 경우, 결과는 111, 112, 122, 222 총 4가지의 경우의 수가 있다. 이는 4개의 영역(공 K개+구분선 N-1개)에 구분선을 어디에 둘 것인가와 동일한 문제이다.

111	1	1	1	/
112	1	1	/	2
122	1	/	2	2
222	/	2	2	2

즉, K+N-1개의 영역에서 N-1개 구분선의 위치를 순서 없이 고르는 것으로 이해할 수 있다. 또한 K+N-1 개의 영역에서 K개 물건의 위치를 순서 없이 고르는 것이라고도 할 수 있다. ($_{K+N-1}C_{N-1} = {}_{K+N-1}C_{(K+N-1)-(N-1)} = {}_{N+K-1}C_K$)
다시 말해, 서로 다른 N가지 물건 중 중복을 허용하여 K개를 택하는 중복조합의 수는 $_{N+K-1}C_K$ (= $_nH_k$)으로 나타낼 수 있다.

조합의 성질(파스칼의 삼각형) : $_{N}C_{K} = {}_{N-1}C_{K-1} + {}_{N-1}C_{K}$

- **수식을 통한 증명**

$$_{N-1}C_{K-1} + {}_{N-1}C_{K} = \frac{(N-1)!}{\big((N-1)-(K-1)\big)!(K-1)!} + \frac{(N-1)!}{((N-1)-K)!K!}$$

$$= \frac{(N-1)!}{(N-K)!(K-1)!} + \frac{(N-1)!}{(N-K-1)!K!}$$

$$= \frac{K(N-1)!}{(N-K)!K!} + \frac{(N-K)(N-1)!}{(N-K)!K!}$$

$$= \frac{K(N-1)! + (N-K)(N-1)!}{(N-K)!K!}$$

$$= \frac{(K+(N-K))(N-1)!}{(N-K)!K!}$$

$$= \frac{N!}{(N-K)!K!} = {}_{N}C_{K}$$

조합의 성질(파스칼의 삼각형) : ${}_{N}C_{K} = {}_{N-1}C_{K-1} + {}_{N-1}C_{K}$

- **조합적 해석을 통한 증명**

1번부터 N번까지 번호가 붙어있는 N개의 사과 중 K개를 고르는 방법을 생각해 보면 크게 두 가지의 경우로 나눌 수 있다.

(1) 1번 사과를 고르는 경우

1번 사과를 이미 골랐으므로, 남은 2번~N번의 N-1개의 사과 중 K-1개의 사과를 골라야 한다.(${}_{N-1}C_{K-1}$)

(2) 1번 사과를 고르지 않는 경우

1번 사과를 제외한 2번~N번의 사과 중 K의 사과를 골라야 한다. (${}_{N-1}C_{K}$)

따라서, N개의 사과 중 K개를 고르는 경우의 수 ${}_{N}C_{K}$는 위의 두 경우를 합한

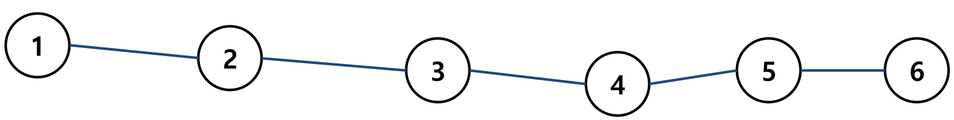
${}_{N-1}C_{K-1}+{}_{N-1}C_{K}$와 같다.

연습문제

1. $\left(x+\frac{1}{x}\right)^{10}$의 전개식에서 $\frac{1}{x^4}$의 계수는?

2. $\binom{5}{1}+\binom{5}{2}+\binom{5}{3}+\binom{5}{4}+\binom{5}{5}$의 값은?

3. 6가지의 물감으로 아래 노드들을 색칠하려 한다. 인접한 노드들은 서로 다른색을 칠해야 한다고 할 때, 1번부터 6번까지의 노드를 색칠하는 방법의 수는?



4. 3번 문제와 같은 구조의 노드들이 주어졌을 때, 10,000,000가지의 물감으로 인접한 노드들을 서로 다른 색으로 색칠하는 방법의 수를 100,000,123으로 나눈 나머지를 구하시오.

연습문제 접근법

(1) $(a+b)^3 = (a+b) \times (a+b) \times (a+b)$으로 생각하여 전개식을 구해본다.

(2) $(1+1)^3 = (1+1) \times (1+1) \times (1+1)$으로 나타낼 수 있음을 생각해 본다.

(3) 한 노드에 칠할 수 있는 색은 인접한 노드에 칠해진 색과 다른 색이므로, 인접한 노드들에 칠해진 색의 종류를 확인하고 경우의 수를 구해본다.

연습문제

1. $\left(x+\frac{1}{x}\right)^{10}$의 전개식에서 $\frac{1}{x^4}$의 계수는?

$\left(x+\frac{1}{x}\right)^{10} = \left(1+\frac{1}{x}\right)\times \ \dots \times \left(1+\frac{1}{x}\right) = \sum_{i=0}^{i=10}\binom{10}{i}x^i \times \left(\frac{1}{x}\right)^{10-i} = \sum_{i=0}^{i=10}\binom{10}{i}(x)^{2i-10}$으로 나타낼 수 있다.

따라서 $\frac{1}{x^4}$의 계수는 $i=3$일 때 이므로 $\binom{10}{3}=120$이다.

2. $(1+1)^5 = \sum_{i=0}^{i=5}\binom{5}{i}1^i \times 1^{5-i} = \sum_{i=0}^{i=5}\binom{5}{i} = \binom{5}{0}+\binom{5}{1}+\binom{5}{2}+\binom{5}{3}+\binom{5}{4}+\binom{5}{5}$임을 알 수 있다.

따라서 $\binom{5}{1}+\binom{5}{2}+\binom{5}{3}+\binom{5}{4}+\binom{5}{5} = 2^5-\binom{5}{0} = 31$이다.

3. 6번 부터 색칠해 볼 때 6번에 칠할 수 있는 색의 수는 6가지,
5번에 칠할 수 있는 색의 수는 6번과 같은 경우를 제외한 5가지,
4번에 칠할 수 있는 색의 수는 5번과 같은 경우를 제외한 5가지,
…
1번에 칠할 수 있는 색의 수는 2번과 다른 5가지이다. 따라서 6×5^5가지 경우가 있다.

연습문제

4. 3번과 마찬가지로 $10000000 \times 9999999^5$ 가지 경우가 있으므로 결과값을 100000123으로 나눈 나머지를 구하면 된다.

하지만$10000000 \times 9999999^5$의 결과값이 2^{63}보다 크므로 다음과 같이 계산한다.

$10000000 \times 9999999^5$

$\equiv 67000246 \times 9999999^4$

$\equiv 68897909 \times 9999999^3$

$\equiv \ldots$

$\equiv 25529468 \pmod{100000123}$

이항정리

N이 양의 정수일 때 $(a+b)^n$ 를 전개하면 아래와 같은 식으로 정리된다.

$(a+b)^n = {}_n\mathrm{C}_0 a^n + {}_n\mathrm{C}_1 a^{n-1}b + {}_n\mathrm{C}_2 a^{n-2}b^2 + \dots + {}_n\mathrm{C}_k a^{n-k}b^k + \dots + {}_n\mathrm{C}_n b^n$

이것을 이항정리라 하고 $(a+b)^n$을 전개했을 때 각 항들의 계수를 **이항계수**라고 한다.

$(a+b)^3=(a+b)(a+b)(a+b)$ 를 예로 들어 이항계수를 생각하면, 전개식 중의 a^2b^1의 항은 3개의 인수 $(a+b)$의 각각에서 a 또는 b 중의 어느 하나를 택하되 1개에서는 b를, 나머지 (3 - 1)개에서는 a를 택하여 곱하면 된다.
a^2b^1항은 3개에서 1개의 b를 택하는 경우의 수만큼 나타나므로, 이 항의 계수는 ${}_3\mathrm{C}_1$ (= ${}_3\mathrm{C}_2$) 이다.

따라서 $(a+b)^n$를 전개했을 때 $a^{n-k}b^k$ 항의 계수는 ${}_n\mathbf{C}_k$이다.

이항정리

$(a+b)^n$의 전개식에서 이항계수를 아래와 같이 삼각형으로 배열한 것을 **파스칼의 삼각형**이라고 한다.

조합에서도 증명했듯이 $_{N}C_{K} = {}_{N-1}C_{K-1} + {}_{N-1}C_{K}$ 이 성립한다. 이는 $(a+b)^{n-1}$의 전개식에서 a^{k-1}과 a^{k}항이 등장하는 경우의 각각의 계수 합이 $(a+b)^n$에서 a^k항의 계수와 일치함을 의미한다.

					1					
				1		1				
			1		2		1			
		1		3		3		1		
	1		4		6		4		1	
1		5		10		10		5		1

					$_0C_0$					
				$_1C_0$		$_1C_1$				
			$_2C_0$		$_2C_1$		$_2C_2$			
		$_3C_0$		$_3C_1$		$_3C_2$		$_3C_3$		
	$_4C_0$		$_4C_1$		$_4C_2$		$_4C_3$		$_4C_4$	
$_5C_0$		$_5C_1$		$_5C_2$		$_5C_3$		$_5C_4$		$_5C_5$

조합($_NC_K$)값을 계산하는 여러 방법의 시간복잡도와 유의사항

- **정의($_NC_K=\frac{N!}{(N-K)!K!}$)에 따라 계산하는 방법**

$O(N)$의 시간복잡도를 가지며, 구하고자 하는 최종 $_NC_K$의 값은 자료형(int, long등)의 범위 내에 있다 하더라도 $N!$은 매우 빠른 속도로 증가하기 때문에 계산과정 중 자료형의 범위를 벗어날 수 있음을 유의해야 한다. ($15! > 2^{31} - 1,\ 21! > 2^{63} - 1$)

- **DP와 파스칼의 삼각형을 통해 계산하는 방법**

$D[n][k]$=$_nC_k$ 라고 정의하면, 앞서 배운 조합의 성질(파스칼의 삼각형)을 통해
$D[n][k] = D[n-1][k-1] + D[n-1][k]$임을 알 수 있다.
마찬가지로 $O(N^2)$의 시간복잡도를 가지며, 메모이제이션을 통해 조합 값을 여러 번 구할 때 불필요한 계산을 줄여준다는 장점이 있다.
하지만 DP배열을 만들기 위해 $N \times N$배열만큼의 공간이 필요하기 때문에 N의 값이 클 경우 사용이 불가능하다.

모듈러 연산이 있는 조합($_NC_K$)값을 계산하는 여러 방법의 시간복잡도와 유의사항

- **정의($_NC_K=\frac{N!}{(N-K)!(K!)}$)에 따라 계산하는 방법**

일반적으로 모듈러 연산이 있는 조합 값을 구하는 문제는 조합 값이 매우 커서 이를 모듈러 한 값을 출력하는 문제이다. 따라서 정의에 의해 계산할 경우 $_NC_K$의 값이 자료형의 범위를 벗어나는 경우가 대부분이므로, **계산이 불가능하다**.

이를 피하기 위해 계산과정 중 모듈러를 취할 경우 곱셈에 대해선 분배법칙이 성립하지만 나눗셈에 대해선 이를 보장하지 않기 때문에 마찬가지로 **계산이 불가능하다.**

예1) 17×19를 4로 나눈 나머지(①=②)

$17 \times 19 \equiv 323 \equiv 3\ (mod\ 4)$ -①

$17(mod\ 4) \times 19(mod\ 4) \equiv 1(mod\ 4) \times 3(mod\ 4) \equiv 3(mod\ 4)$ - ②

예2) 150÷6를 10으로 나눈 나머지(①≠②)

$150 \div 6 \equiv 25 \equiv 5\ (mod\ 10)$ - ①

$150(mod\ 10) \div 6(mod\ 10) \equiv 0(mod\ 10) \div 6(mod\ 10) \equiv 0\ (mod\ 10)$ - ②

모듈러 연산이 있는 조합($_NC_K$)값을 계산하는 여러 방법의 시간복잡도와 유의사항

- **DP와 파스칼의 삼각형을 통해 계산하는 방법**

$D[n][k] \equiv {}_nC_k \ (mod\ p)$라고 정의하면

$$D[n][k] \equiv {}_nC_k \ (mod\ p)$$
$$\equiv ({}_{n-1}C_{k-1} + {}_{n-1}C_k)(mod\ p)$$
$$\equiv ({}_{n-1}C_{k-1})(mod\ p) + ({}_{n-1}C_k)(mod\ p)$$
$$\equiv D[n-1][k-1] + D[n-1][k]$$

$O(N^2)$의 시간복잡도를 가지며, 메모이제이션을 통해 조합 값을 여러 번 구할 때 불필요한 계산을 줄여준다는 장점이 있다. 하지만 앞서 언급한 것 처럼 DP배열을 만들기 위해 $N \times N$배열만큼의 공간이 필요하므로 N의 값이 클 경우 사용이 불가능하다.

사전

동호와 규완이는 212호에서 문자열에 대해 공부하고 있다. 김진영 조교는 동호와 규완이에게 특별 과제를 주었다. 특별 과제는 특별한 문자열로 이루어 진 사전을 만드는 것이다. 사전에 수록되어 있는 모든 문자열은 N개의 "a"와 M개의 "z"로 이루어져 있다. 그리고 다른 문자는 없다. 사전에는 알파벳 순서대로 수록되어 있다.

규완이는 사전을 완성했지만, 동호는 사전을 완성하지 못했다. 동호는 자신의 과제를 끝내기 위해서 규완이의 사전을 몰래 참조하기로 했다. 동호는 규완이가 자리를 비운 사이에 몰래 사전을 보려고 하기 때문에, 문자열 하나만 찾을 여유밖에 없다.

N과 M이 주어졌을 때, 규완이의 사전에서 K번째 문자열이 무엇인지 구하는 프로그램을 작성하시오.

[입력]

첫째 줄에 N, M, K가 순서대로 주어진다. N과 M은 100보다 작거나 같은 자연수이고, K는 1,000,000,000보다 작거나 같은 자연수이다.

[출력]

첫째 줄에 규완이의 사전에서 K번째 문자열을 출력한다. 만약 규완이의 사전에 수록되어 있는 문자열의 개수가 K보다 작으면 -1을 출력한다.

사전

(1) ‘a’가 N개, ‘z’가 M개인 문자열 중 첫 문자가 ‘a’인 경우와 ‘z’인 경우의 문자열이 각각 몇 개 씩 존재하는지 생각해 보자.

(2) 사전순으로 X번째 문자열의 첫 문자는 무엇일지 생각해보자.

실전문제

(1) 첫 문자열이 'a' 혹은 'z'인 문자열의 수

길이가 N+M이고 'a'의 개수가 N개, 'z'의 개수가 M개인 문자열은 첫 문자가 'a'인 경우와 'z'인 경우밖에 없다.

첫 문자가 'a'인 경우 첫 문자가 'a'으로 결정되었으므로 총 길이 N + M – 1 개 중 M개가 'z'인 문자열의 수와 같다. 즉, 첫 문자가 'a'인 문자열의 수는 ${}_{N+M-1}C_M$ 이다.

첫 문자가 'z'인 경우 첫 문자가 'z'로 결정되었으므로 총 길이 N + M – 1 개 중 M – 1개가 'z'인 문자열의 수와 같다. 즉, 첫 문자가 'z'인 문자열의 수는 ${}_{N+M-1}C_{M-1}$ 이다.

위에 내용을 정리하면 길이가 N + M이고 'z'의 개수가 M개인 문자열의 수는 다음과 같다.

=> ${}_{N+M}C_M = {}_{N+M-1}C_M + {}_{N+M-1}C_{M-1}$

실전문제

(2) K번째 문자열 찾기

앞에서 정리한 내용으로 사전에 들어갈 문자열은 첫 문자가 'a'인 경우와 'z'인 경우의 합으로 나타낼 수 있으며 수식으로 나타내면 ${}_{N+M}C_{M} = {}_{N+M-1}C_{M} + {}_{N+M-1}C_{M-1}$ 이다. 또한 문자열은 사전 순으로 나타낼 것이므로 순서적으로 첫 문자가 'a'인 모든 비트문자열은 첫 문자가 'z'인 문자열 앞에 있다.

만약 K번째 문자열의 첫 문자가 'a'이라면 $K <= {}_{N+M-1}C_{M}$ 임을 만족해야 한다. K가 첫 문자를 'a'으로 만들 수 있는 문자열의 수보다는 작거나 같아야 하기 때문이다.

이 성질을 이용하면 첫 문자를 결정할 수 있으며, 두 번째 문자를 결정하는 문제는 전체 길이가 N+M-1 이고 'z'의 개수가 M 개인 문제로 놓고 똑같은 원리로 풀 수 있다.

실전문제

순서	문자열
1	a a z z z
2	a z a z z
3	a z z a z
4	a z z z a
5	z a a z z
6	z a z a z
7	z a z z a
8	z z a a z
9	z z a z a
10	z z z a a

N이 2이고 M이 3인 문자열의 경우이다.

K가 7인 경우를 생각해 보자.

가장 첫 문자가 'a'인 문자열의 수는 ${}_4C_3$ 이므로 4개가 있다는 것을 오른쪽 표를 이용하여 쉽게 구할 수 있다.

우리가 찾는 K는 7이므로 첫 문자는 'z'임을 알 수 있다.

첫 문자가 'z'로 결정되면, 첫 문자가 'a'인 문자열을 건너뛰고

찾게 되므로 첫 문자가 'z'로 시작하는 문자열 중 3번째인 문자열을 찾으면 된다.

즉, 두 번째 문자를 결정하는 문제는 전체 길이가 4이고 M이 2인 문자열 중 K가 3인 문제를 푸는 것과 동일한 문제가 된다.

n \ k	1	2	3	4	5
1	1				
2	2	1			
3	3	3	1		
4	4	6	4	1	
5	5	10	10	5	1

<nCk>

Summary

- 조합공식을 이용하여 여러 경우의 수를 구한다.
- 순열, 조합에 중복이 발생하는 경우 이를 제외하고 경우의 수를 구할 수 있다.
- 이항정리를 이해하고 이항계수를 조합의 성질을 활용하여 나타낼 수 있다.
- 파스칼의 삼각형을 활용하여 $_{N}C_{K}$ 의 값을 계산할 수 있다.