SAMSUNG
SW
ACADEMY
FOR
YOUTH

함께가요 미래로!
Enabling People

# 삼성 청년
# SW 아카데미

APS 응용

# 목차

1. 동적 계획법 기본 - 이항 계수 구하기
2. 동적 계획법 기본 - 거스름돈 구하기

SAMSUNG
SW
ACADEMY
FOR
YOUTH

함께가요 미래로!
Enabling People

# 동적 계획법 기본 - 이항 계수 구하기

# 문제 제시 : 계수 값 구하기

**다음 수식의 ?의 값은?**

- $(x + y)^4 = x^4 + 4x^3y + ?x^2 y^2 + 4xy^3 + y^4$
- $(x + y)^n$ 을 전개 했을 때 $x^k y^{n-k}$의 값은?

# 이항 계수 구하기

## 이항정리는

- 이항 다항식 $x + y$의 거듭제곱 $(x + y)^n$에 대해서, 전개한 각 항 $x^k y^{n-k}$의 계수 값을 구하는 정리이다.
- 구체적으로 $x^k y^{n-k}$의 계수는 n개에서 k개를 고르는 조합의 가짓수인 ${}_nC_k$이고 이를 이항계수라고 부른다.
- 예를 들어, $n = 2$, $n = 3$, 그리고 $n = 4$일 경우에는 다음과 같다.
  - $(x + y)^2 = x^2 + 2xy + y^2$
  - $(x + y)^3 = x^3 + 3x^2y + 3xy^2 + y^3$
  - $(x + y)^4 = x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4$

# 이항 계수 구하기

- 이항계수 구하는 공식

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \text{ for } 0 \le k \le n$$

- 계산량이 많은 n!이나 k!을 계산하지 않고 이항계수를 구하기 위해서 통상 다음 수식을 사용한다.

$$\binom{n}{k} = \begin{cases} \binom{n-1}{k-1} + \binom{n-1}{k} & \text{if } 0 < k < n \\ 1 & \text{if } k = 0 \text{ or } k = n \end{cases}$$

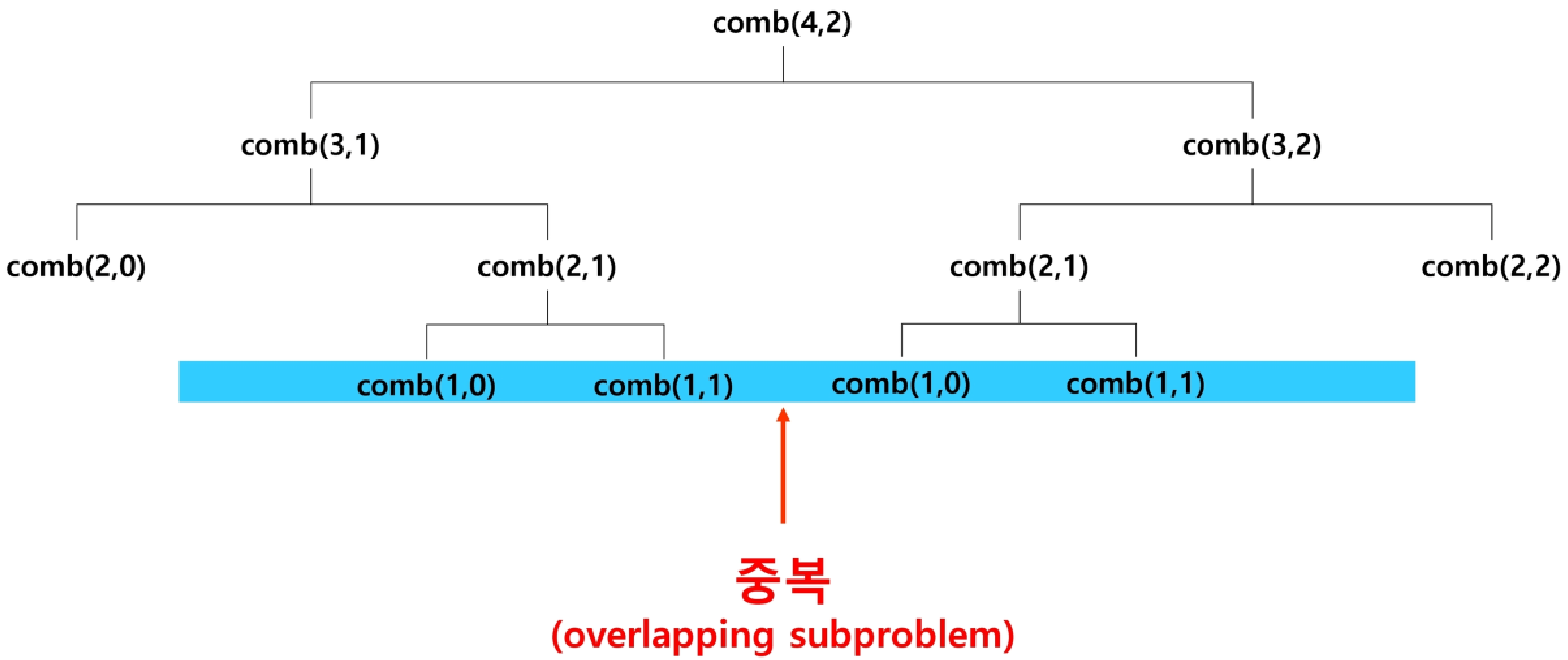
# 이항 계수 구하기

- 이항계수 nCk 를 구하는 재귀 함수

```
comb(n,k)
      IF n==k or k==0 : RETURN 1
      ELSE : RETURN comb(n-1,k-1) + comb(n-1,k)
```

# 이항 계수 구하기

- **Call Tree**



# 이항 계수 구하기

## 파스칼의 삼각형

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 1 | | | | |
| | | | 1 | | 1 | | | |
| | | 1 | | 2 | | 1 | | |
| | 1 | | 3 | | 3 | | 1 | |
| 1 | | 4 | | 6 | | 4 | | 1 |
| | | | | … | | | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | c(0,0) | | | | |
| | | | c(1,0) | | c(1,1) | | | |
| | | c(2,0) | | c(2,1) | | c(2,2) | | |
| | c(3,0) | | c(3,1) | | c(3,2) | | c(3,3) | |
| c(4,0) | | c(4,1) | | c(4,2) | | c(4,3) | | c(4,4) |
| | | | | … | | | | |

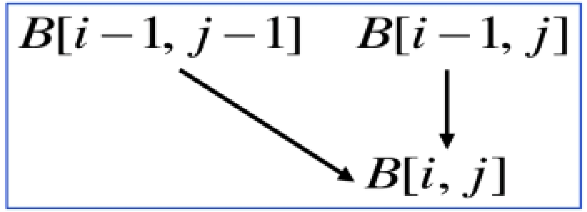
# 이항 계수 구하기

- 동적 계획법을 적용한 이항계수 계산
- O(nk)

```
bino(n, k)
 B[][]
 FOR i in 0 → n
   FOR j in 0 → minimum(i, k)
     IF j = 0 OR j = i
       B[i][j] ← 1
     ELSE
       B[i][j] ← B[i-1][j-1] + B[i-1][j]

  RETURN B[n][k]
```

| | 0 | 1 | 2 | 3 | 4 | $j$ | $k$ |
|---|---|---|---|---|---|---|---|
| 0 | 1 | | | | | | |
| 1 | 1 | 1 | | | | | |
| 2 | 1 | 2 | 1 | | | | |
| 3 | 1 | 3 | 3 | 1 | | | |
| 4 | 1 | 4 | 6 | 4 | 1 | | |
| $i$ | | | | | | | |
| $n$ | | | | | | | |



SAMSUNG
SW
ACADEMY
FOR
YOUTH

함께가요 미래로!
Enabling People

# 동적 계획법 기본 - 동전 거스름돈 구하기

# 문제 제시 : 동전 거스름 돈 구하기

- **동전의 종류**
  - 1원, 4원, 6원

- **8원을 거슬러주려 한다. 최소 몇 개의 동전을 거슬러 주면 되나?**

# 문제 제시 : 동전 거스름 돈 구하기

- **그리디 방법의 접근**
  - 6원, 1원, 1원
- **최적은?**
  - 4원, 4원

- **그리디 방법이 항상 최적해를 구하는 것은 아니다. 어떻게 풀어야 하나?**
  - 동적 계획법으로 접근해 보자.

# 문제 제시 : 동전 거스름 돈 구하기

- **우선 재귀적인 8원 잔돈에 대한 알고리즘**
  - 3가지 동전 각각을 선택해서 재귀적으로 해결

1원 동전 한 개 + <u>7원에 대한 최적해</u>
4원 동전 한 개 + 4원에 대한 최적해
6원 동전 한 개 + 2원에 대한 최적해
위의 3가지 해 중 최적해를 선택

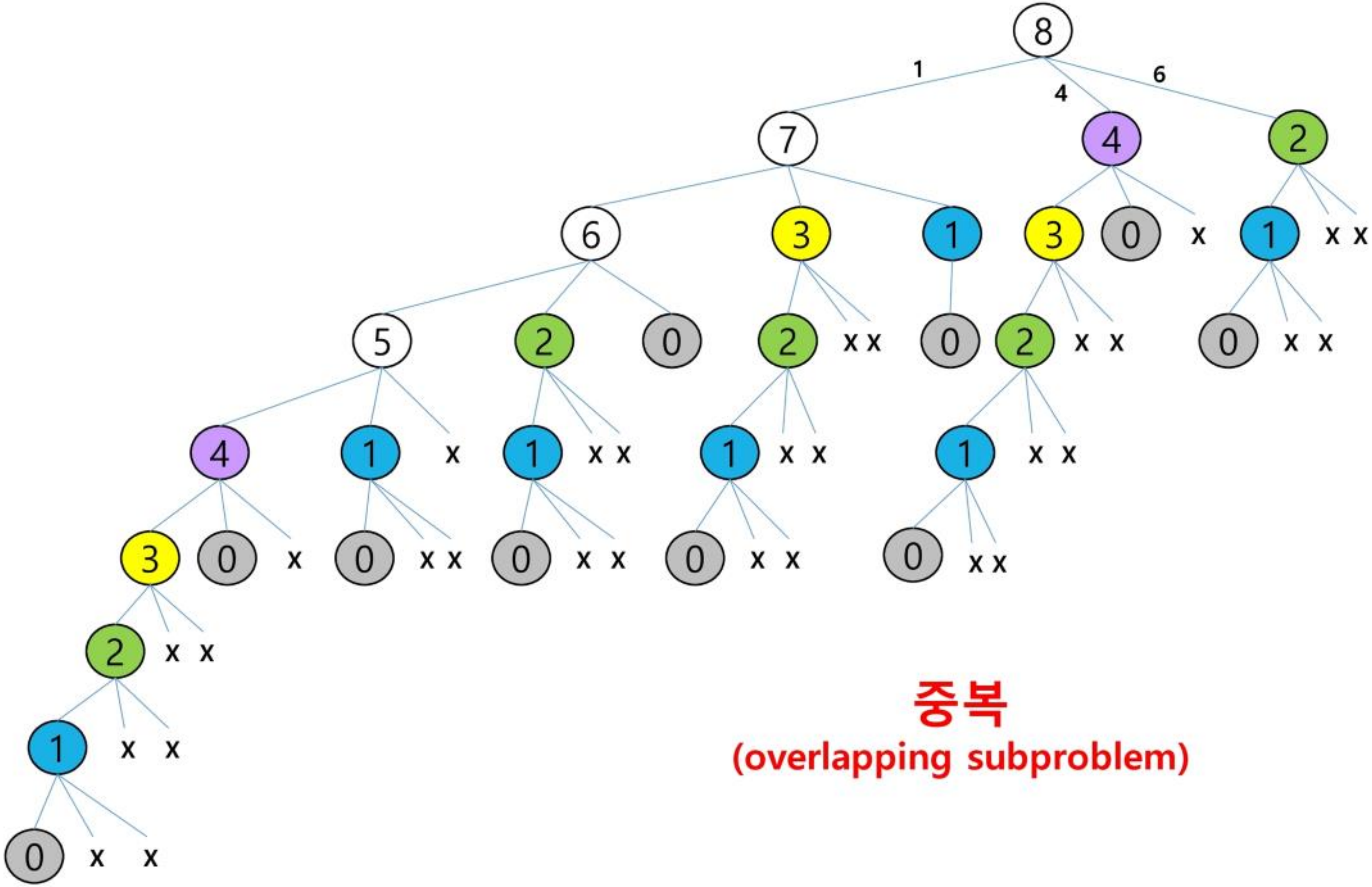
*7원에 대한 최적해는 다시 1원, 4원, 6원 동전을 선택하고 나머지 액수에 대한 최적해*

1원 동전 한 개 + <u>6원에 대한 최적해</u>
4원 동전 한 개 + 3원에 대한 최적해
6원 동전 한 개 + 1원에 대한 최적해
위의 3가지 해 중 최적해를 선택

…

# 문제 제시 : 동전 거스름 돈 구하기

- **Call tree**




**중복**
**(overlapping subproblem)**

# 문제 제시 : 동전 거스름 돈 구하기

## DP 접근 : 상향식

- 1원에 대한 최적해 → (선택) 2원에 대한 최적해 → (선택) 3원에 대한 최적해 → (선택) 4원에 대한 최적해 → (선택) ...
- C[n] = n원을 거슬러 줄 때의 최적
- 점화식 : C[n] = MIN ( n-1>=0➔ C[n-1]+1 , n-4>=0 ➔ C[n-4]+1 , n-6>=0 ➔ C[n-6]+1 )

| n | choice | C[n] |
|---|---|---|
| 0 | 0 | 0 |
| 1 | C[n - 1] + 1 → 1 | 1 |
| 2 | C[n - 1] + 1 → 2 | 2 |
| 3 | C[n - 1] + 1 → 3 | 3 |
| 4 | C[n - 1] + 1, C[n - 4] + 1 = MIN (4, 1) → 1 | 1 |
| 5 | C[n - 1] + 1, C[n - 4] + 1 = MIN (2, 2) → 2 | 2 |
| 6 | C[n - 1] + 1, C[n - 4] + 1, C[n - 6] + 1 = MIN (3, 3, 1) → 1 | 1 |
| 7 | C[n - 1] + 1, C[n - 4] + 1, C[n - 6] + 1 = MIN (2, 4, 2) → 2 | 2 |
| 8 | C[n - 1] + 1, C[n - 4] + 1, C[n - 6] + 1 = MIN (3, 2, 3) → 2 | 2 |

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로! Enabling People

# 다음 방송에서 만나요!

삼성 청년 SW 아카데미