

백준 12865, 평범한 배낭

키워드 : 냅색 문제

Input

w : 가방 무게, V : 가방 가치

	0	1	2	3
w	6	4	3	5
V	13	8	6	12

$DP[i][j]$: NE 가방에 들어가는 물건 수, i 는 가방에 들여갈 수 있는 무게

	1	2	3	4	5	6	7
6							
4							
3							
5							

Step 1

먼저 무게 6인 물건을 넣었을 때 가치를 채운다

	1	2	3	4	5	6	7
6	0	0	0	0	0	13	13
4							
3							
5							

Step 2

그 다음 무게가 4인 물건의 가치를 넣는다,

이때 $dp[무게][i-1]$ 값을 대입 후, $dp[무게][i]$ 와 비교하여 값을 바꾼다

	1	2	3	4	5	6	7
6	0	0	0	0	0	13	13
4	0	0	0	8	8	13	13
3							
5							

Step 3

그 다음 무게가 3인 물건의 가치를 넣는다,

이때 $dp[무게][i-1]$ 값을 대입 후, $dp[무게][i]$ 와 비교하여 값을 바꾼다

	1	2	3	4	5	6	7
6	0	0	0	0	0	13	13
4	0	0	0	8	8	13	13
3	0	0	6	8	8	13	14
5							

Step 4

마지막 무게가 5인 물건의 가치를 넣는다,

이때 $dp[무게][i-1]$ 값을 대입 후, $dp[무게][i]$ 와 비교하여 값을 바꾼다

	1	2	3	4	5	6	7
6	0	0	0	0	0	13	13
4	0	0	0	8	8	13	13
3	0	0	6	8	8	13	14
5	0	0	6	8	12	13	14

백준 12865, 평범한 배낭

키워드 : 문제 이해 부족, 물건을 여러개 담을 수 있는 것으로 이해

Input W: 가방 무게, V: 물건 가격

	0	1	2	3
W	6	4	3	5
V	13	8	6	12

dp[i][k] : 가방에 무게 k 만큼 넣었을 때 최대 가치

1	2	3	4	5	6	7
0	0	0	0	0	0	0

Step 1

$k=1$: 무게의 모든 무게가 1보다 크므로 0을 대입

$k=2$: 무게의 모든 무게가 2보다 크므로 0을 대입

$k=3$: 모든 가짐을 무게가 3인 가능한 경우 수 셀고, 가치는 6
가능한 경우 수 있는 경우 확장판 $k=3$, 가치는 0
무게가 3이면 가치가 없으므로 0을 더해 6을 대입

1	2	3	4	5	6	7
0	0	6	0	0	0	0

Step 2

$k=4$,

case1: 무게가 3인 가짐을 넣을 때 가치 6과

$4+2=6$ 이며 dp[4][6]으로 확장된 가치는 6

case2: 무게가 4인 가짐을 넣을 때 가치는 8이고

$4+2=6$ 이며 dp[4][8]으로 확장된 가치는 8

case3: 모든 가짐을 무게가 3인 가능한 경우 수 셀고, 가치는 8

\Rightarrow case1과 case2 경우가 가치가 크므로 8을 대입

1	2	3	4	5	6	7
0	0	6	8	0	0	0

Step 4

$k=5$, case1: 무게가 3인 가짐을 넣을 때 가치 6과

$5+2=7$ 이며 dp[5][7]으로 확장된 가치는 7

case2: 무게가 4인 가짐을 넣을 때 가치는 8이고

$5+2=7$ 이며 dp[5][8]으로 확장된 가치는 8

case3: 무게가 5인 가짐을 넣을 때 가치는 12이고

$5+2=7$ 이며 dp[5][12]으로 확장된 가치는 12

case4: 무게가 6인 가짐을 넣을 때 가치는 13이고

$5+2=7$ 이며 dp[5][13]으로 확장된 가치는 13

\Rightarrow case4 경우가 가치가 제일 크므로 13 대입

1	2	3	4	5	6	7
0	0	6	8	12	13	0

Step 5

$k=6$,

case1: 무게가 3인 가짐을 넣을 때 가치 6과

$6+2=8$ 이며 dp[6][8]으로 확장된 가치는 8

case2: 무게가 4인 가짐을 넣을 때 가치는 8이고

$6+2=8$ 이며 dp[6][8]으로 확장된 가치는 8

case3: 무게가 5인 가짐을 넣을 때 가치는 12이고

$6+2=8$ 이며 dp[6][12]으로 확장된 가치는 12

case4: 무게가 6인 가짐을 넣을 때 가치는 13이고

$6+2=8$ 이며 dp[6][13]으로 확장된 가치는 13

\Rightarrow case4 경우가 가치가 제일 크므로 13 대입

1	2	3	4	5	6	7
0	0	6	8	12	13	0

Step 3

$k=5$,

case1: 무게가 3인 가짐을 넣을 때 가치 6과

$5+2=7$ 이며 dp[5][7]으로 확장된 가치는 7

case2: 무게가 4인 가짐을 넣을 때 가치는 8이고

$5+2=7$ 이며 dp[5][8]으로 확장된 가치는 8

case3: 무게가 5인 가짐을 넣을 때 가치는 12이고

$5+2=7$ 이며 dp[5][12]으로 확장된 가치는 12

\Rightarrow case1과 case2 경우가 가치가 제일 크므로 7 대입

1	2	3	4	5	6	7
0	0	6	8	12	0	0

백준 2655, 가장 높은 탑 쌓기

Step 1

입력 (S: 빙면 높이, H: 높이, W: 너비,
pos: 정렬 순 차례)

	0	1	2	3	4
S	25	4	9	16	1
H	3	4	2	2	5
W	4	6	3	5	2
pos	0	1	2	3	4

입력 높이를
내림차순으로 정렬

정렬 후 (앞으로 블록을 쌓을 때, 높이만 바뀐다.)

	0	1	2	3	4
S	25	16	9	4	1
H	3	2	2	4	5
W	4	5	3	6	2
pos	0	3	2	1	4

dp

maxH : 해당 블록을 제일 첫째 놓았을 때, 힙에 최대 높이
prev : 해당 블록을 바로 앞서 놓은 블록의 index
pos : 해당 블록의 현재 위치

	0	1	2	3	4
maxH	0	0	0	0	0
prev	-1	-1	-1	-1	-1
pos	0	0	0	0	0

Step 2

	0	1	2	3	4
S	25	16	9	4	1
H	3	2	2	4	5
W	4	5	3	6	2
pos	0	3	2	1	4

	0	1	2	3	4
maxH	3	0	0	0	0
prev	-1	-1	-1	-1	-1
pos	0	0	0	0	0

0번 블록은 끝에 놓을 것으 둘 4 번째로
maxH에 3, pos에 0, prev에 -1 대입

Step 3

	0	1	2	3	4
S	25	16	9	4	1
H	3	2	2	4	5
W	4	5	3	6	2
pos	0	3	2	1	4

	0	1	2	3	4
maxH	3	2	0	0	0
prev	-1	-1	-1	-1	-1
pos	0	3	0	0	0

X
1번 블록
0번 블록

1번 블록은 0번보다 높아서 4개므로
0번을 끝에 둘 수 없어서 바로 대입,
현재 위치 3로 pos에 대입

Step 4

	0	1	2	3	4
S	25	16	9	4	1
H	3	2	2	4	5
W	4	5	3	6	2
pos	0	3	2	1	4

	0	1	2	3	4
maxH	3	2	5	0	0
prev	-1	-1	0	-1	-1
pos	0	3	2	0	0

O
2번 블록
0번 블록
1번 블록

2번 블록은 0과 1 블록 끝에 놓을 수 있고
dp[0], maxH에 3과 dp[0], maxH에 대입
dp[2], maxH에 5와 dp[2], maxH에 대입
pos에 현재 위치 2를 대입
prev에는 바로 끝의 블록의 index인 0을 대입

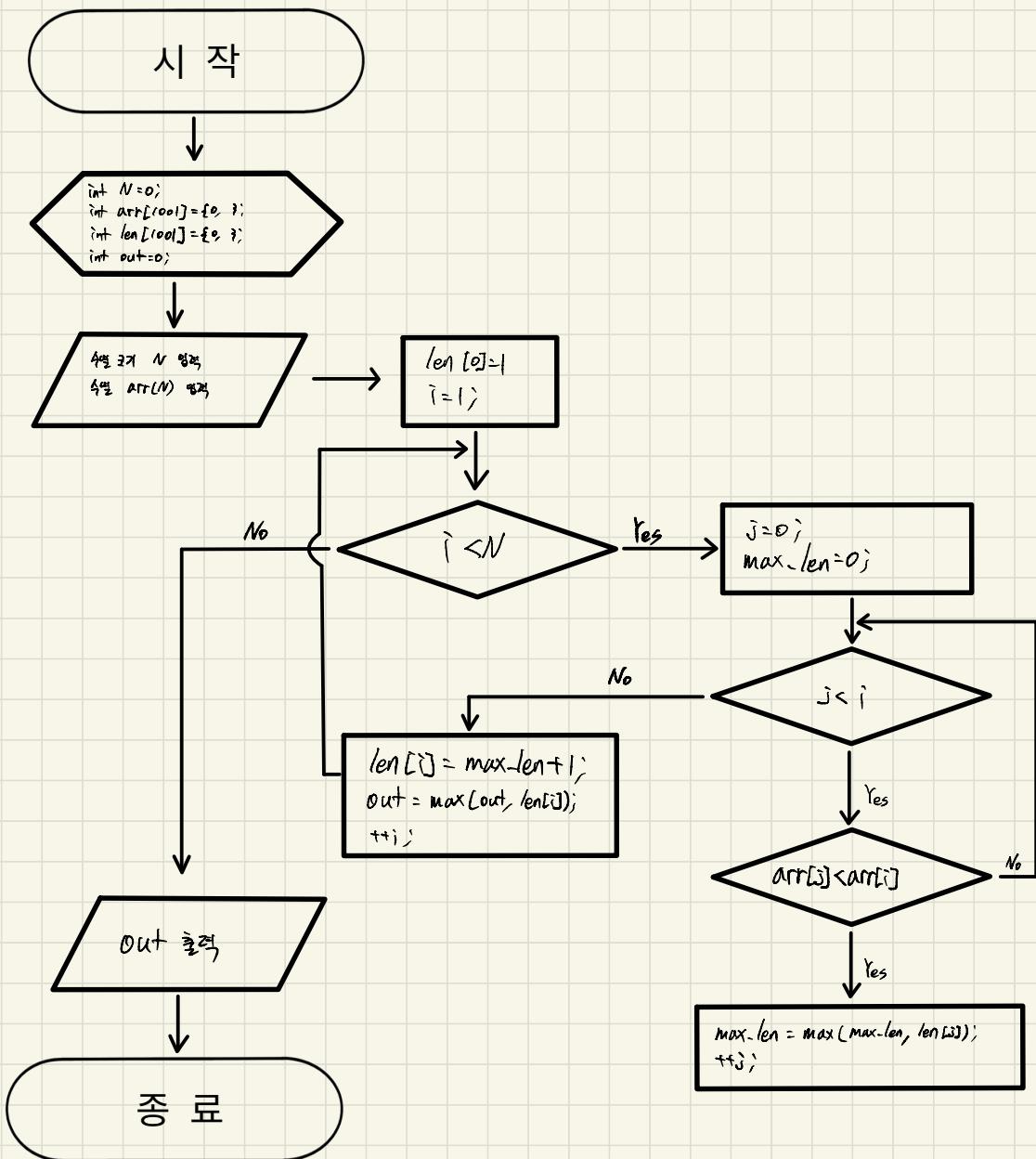
Step 5 4단계 수행 결과

	0	1	2	3	4
maxH	3	2	5	4	10
prev	-1	-1	0	-1	2
pos	0	3	2	1	4

Step 6 출력

```
pseudo code
while (true)
{
    printf("%d\n", cur.dp.pos);
    if (cur.dp.prev == -1)
    {
        break;
    }
    cur.dp = dp[cur.dp.prev];
}
```

백준 11053, 가장 긴 증가하는 수열



백준 1904, 01타일

$$a_1 = 1 \quad \boxed{1}$$

$$a_2 = 2 \quad \boxed{1} \quad \boxed{1} \quad \boxed{0} \quad \boxed{0}$$

$$a_3 = 3 \quad \boxed{} \quad \boxed{1} \quad + \quad \boxed{} \quad \boxed{0} \quad \boxed{0}$$

$\downarrow \quad \quad \quad \downarrow$
 $a_1 \quad + \quad a_2$

$$a_4 = 5 \quad \boxed{} \quad \boxed{1} \quad + \quad \boxed{} \quad \boxed{0} \quad \boxed{0}$$

$\downarrow \quad \quad \quad \downarrow$
 $a_3 \quad + \quad a_2$

$$\text{정의: } a_n = a_{n-1} + a_{n-2}$$

백준 16395, 파스칼의 삼각형

$$\text{정의: } nCr = n-rCr + n-rCr-1$$

pseudo code

$$arr[0][0] = 1;$$

$$arr[1][0] = 1;$$

$$arr[2][0] = 1;$$

for (int i=2; i < N; ++i)

{

 for (int j=0; j < N; ++j)

{

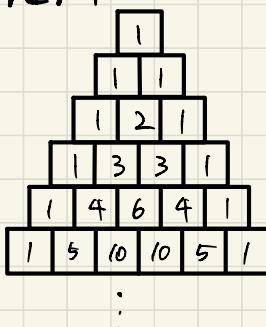
 if (j == 0 || j == N)

 arr[i][j] = 1;

 arr[i][j] = arr[i-1][j] + arr[i-1][j-1];

}

}



:

