

다이나믹 프로그래밍 4 (연습)

최백준 choi@startlink.io

상태 다이나믹

발전소

<https://www.acmicpc.net/problem/1102>

- 발전소를 고치는 방법은 간단하다.
- 고장나지 않은 발전소를 이용해서 고장난 발전소를 재시작하면 된다.
- 하지만, 이 때 비용이 발생한다.
- 이 비용은 어떤 발전소에서 어떤 발전소를 재시작하느냐에 따라 다르다.
- 적어도 P개의 발전소가 고장나 있지 않도록, 발전소를 고치는 비용의 최솟값을 구하는 프로그램을 작성하시오.
- $N \leq 16$

발전소

<https://www.acmicpc.net/problem/1102>

- $D[i]$ = 발전소의 상태를 i 로 만들기 위해 필요한 최소 비용
- i = 발전소를 이진수로 나타낸 상태 (1: 켜있음)
- $D[i \mid (1 \ll k)] = D[i] + A[j][k]$
- j 는 i 에서 켜져있는 발전소
- k 는 i 에서 꺼져있는 발전소

발전소

<https://www.acmicpc.net/problem/1102>

- 소스: <http://codeplus.codes/6cea3e34446e4f5c9f576ab313be1e76>

007

<https://www.acmicpc.net/problem/3056>

- N명의 사람이 미션 N개를 해야 한다. $1 \leq N \leq 20$
- i번 사람이 j번 미션을 했을 때, 성공 확률: $P[i][j]\%$
- 각 사람이 1개의 미션, 각 미션도 1명만 해야 할 때, 확률의 최댓값을 구하는 문제

007

<https://www.acmicpc.net/problem/3056>

- $D[S]$ = 배정된 미션의 상태가 S 일 때, 성공적으로 마칠 확률

007

<https://www.acmicpc.net/problem/3056>

- $D[S]$ = 배정된 미션의 상태가 S 일 때, 성공적으로 마칠 확률
- 앞에서 부터 차례대로 어떤 미션을 선택할지 결정한다.
- i 번 사람이 미션 k 를 선택했다면
- $D[S] = \max(D[S | (1 \leq k)])$

007

<https://www.acmicpc.net/problem/3056>

- 소스: <http://codeplus.codes/aa409b4f4e844036a37ad00df882e651>

타일 채우기

10

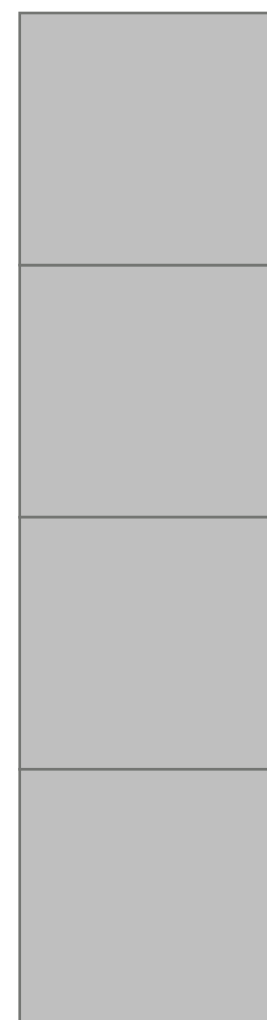
<https://www.acmicpc.net/problem/2718>

- $4 \times N$ 크기의 타일을 1×2 , 2×1 크기의 도미노로 채우는 방법의 수

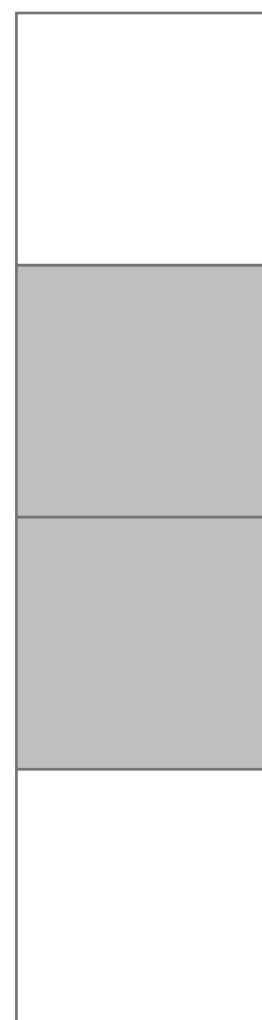
타일 채우기

<https://www.acmicpc.net/problem/2718>

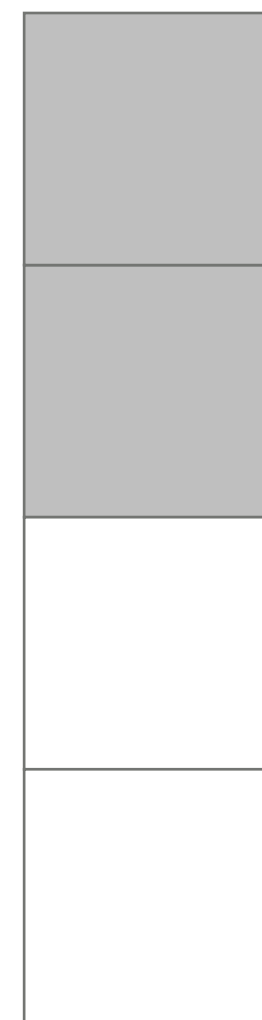
- $F[N] = 4 \times N$ 크기의 타일을 채우는 방법의 수
- $G[N] = 4 \times N$ 크기의 타일을 채우는 방법의 수 (오른쪽-위, 오른쪽-아래는 채우지 않음)
- $H[N] = 4 \times N$ 크기의 타일을 채우는 방법의 수 (오른쪽-아래칸 두 개는 채우지 않음)
- $H[N] = 4 \times N$ 크기의 타일을 채우는 방법의 수 (오른쪽-위칸 두 개는 채우지 않음)



$F[N]$



$G[N]$



$H[N]$



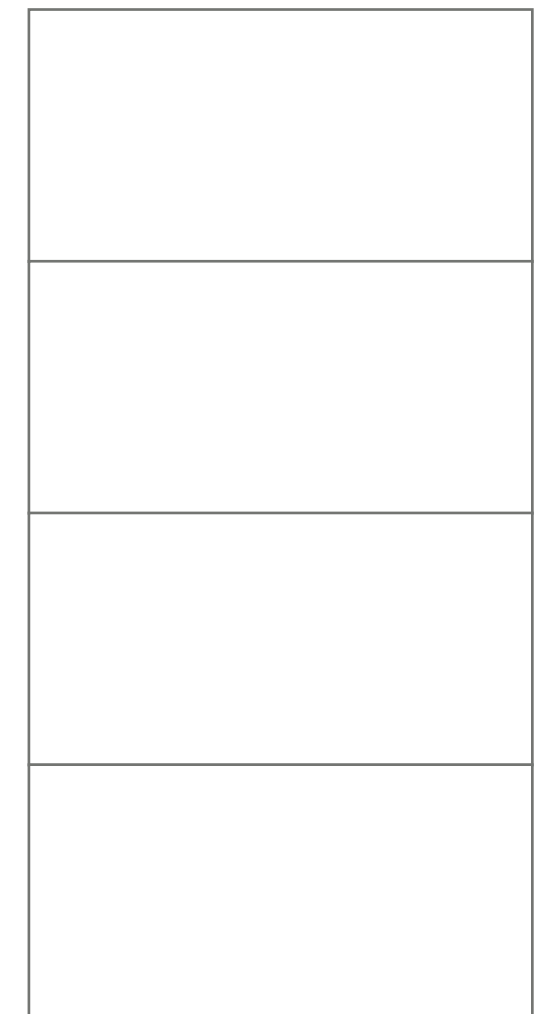
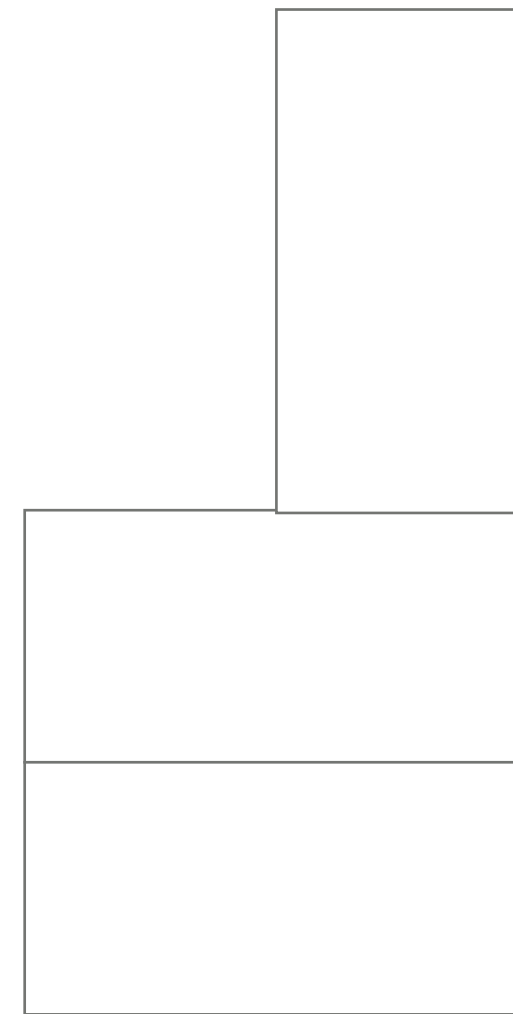
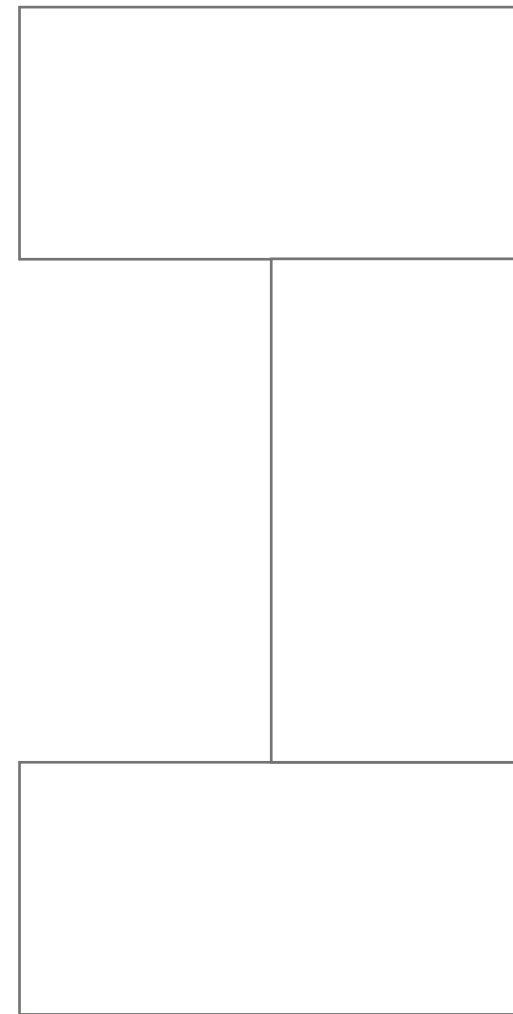
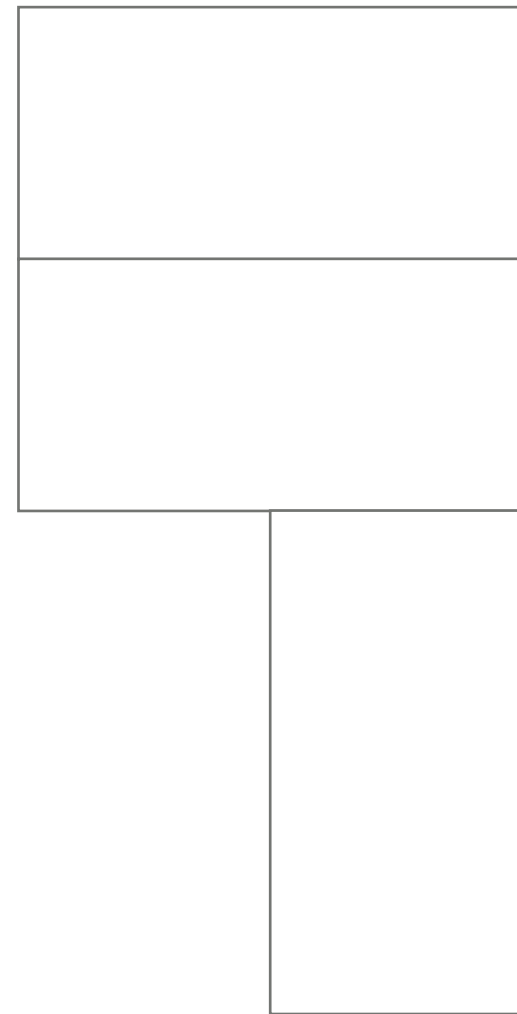
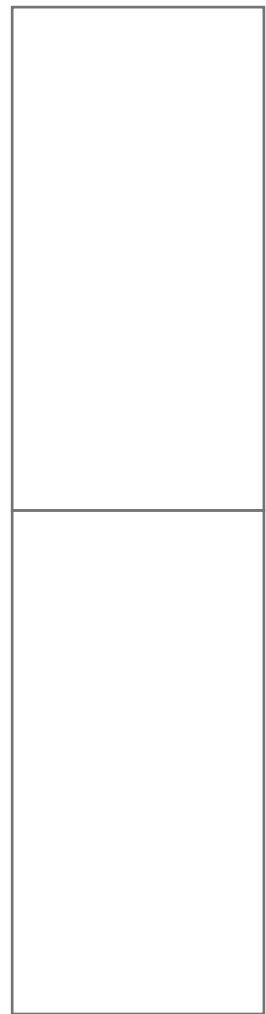
$H[N]$

타일 채우기

12

<https://www.acmicpc.net/problem/2718>

- 가장 오른쪽에 올 수 있는 경우의 수

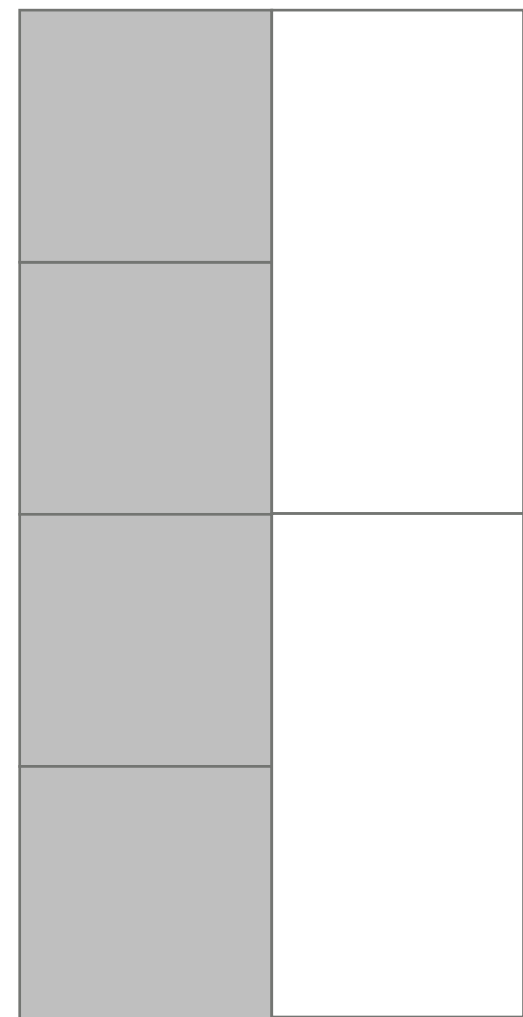


타일 채우기

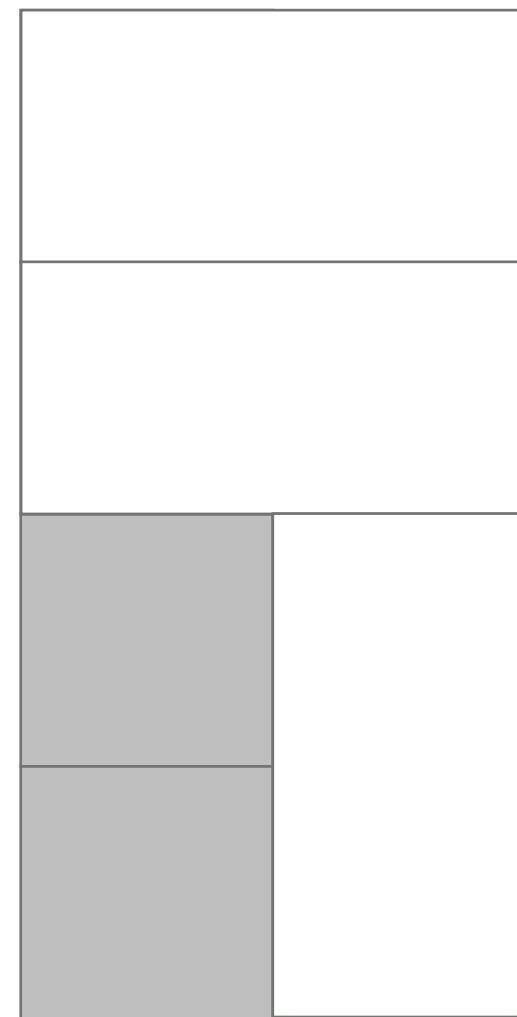
13

<https://www.acmicpc.net/problem/2718>

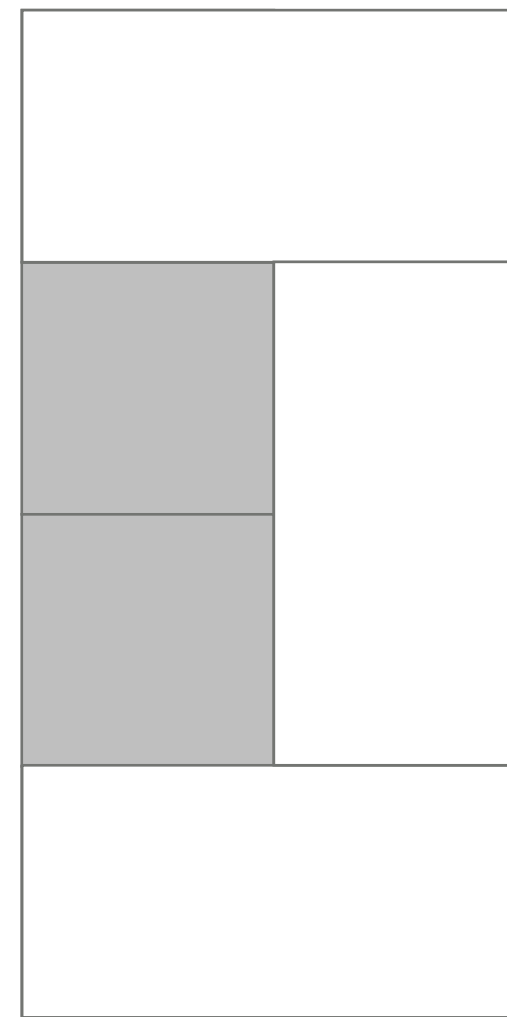
- $F[N] = F[N-1] + H[N-1]*2 + G[N-1] + F[N-2]$



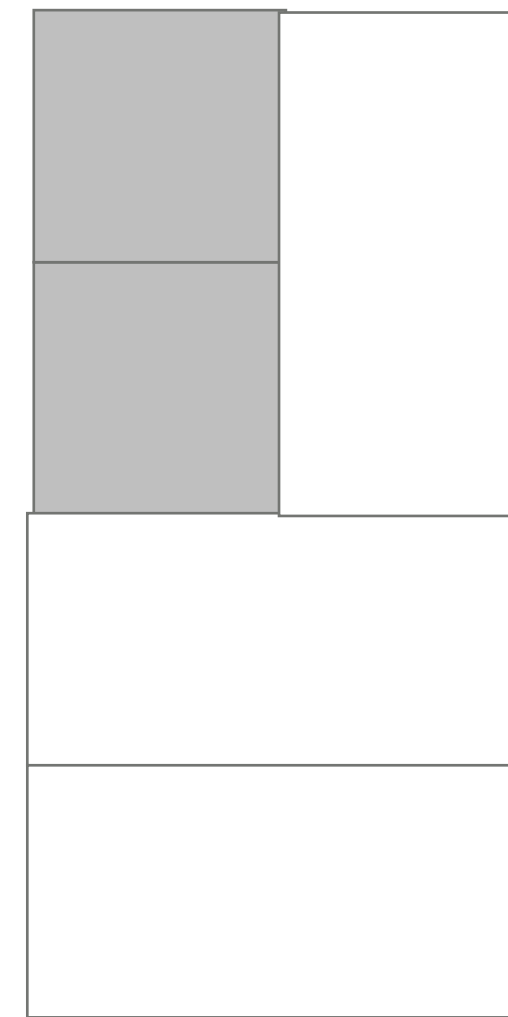
F[N-1]



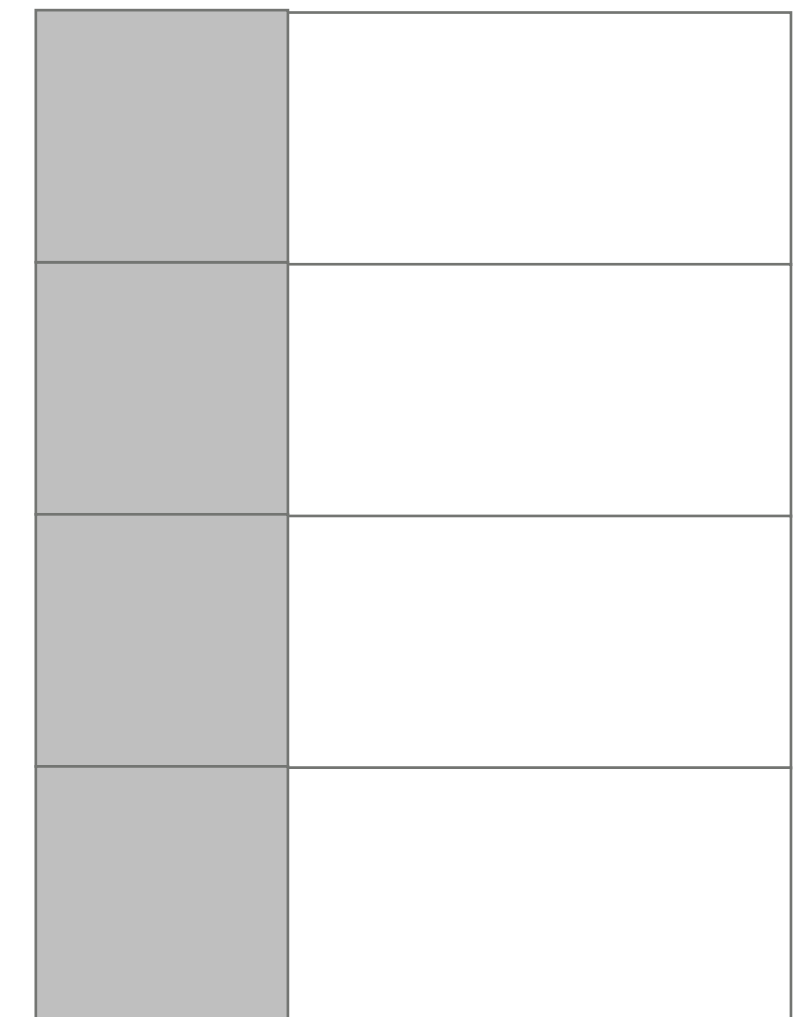
H[N-1]



G[N-1]



H[N-1]



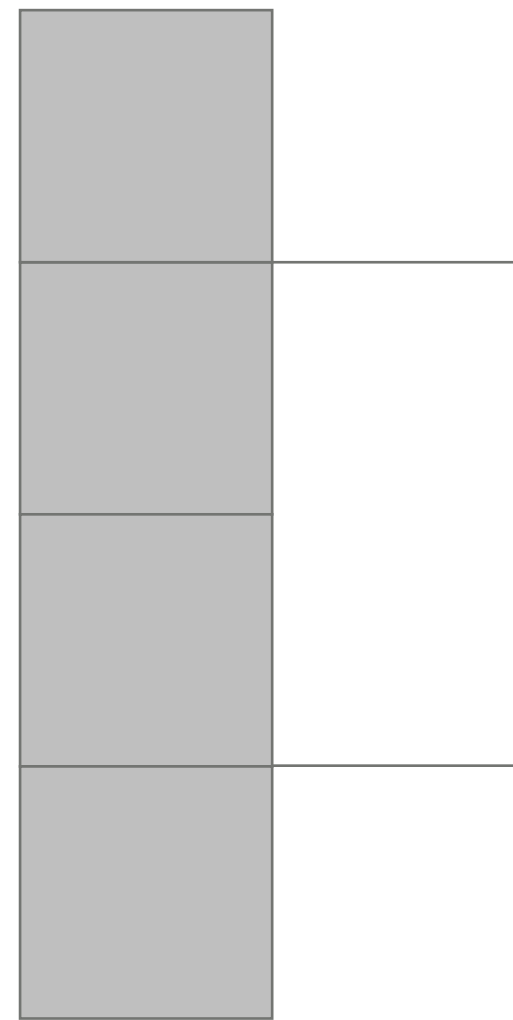
F[N-2]

타일 채우기

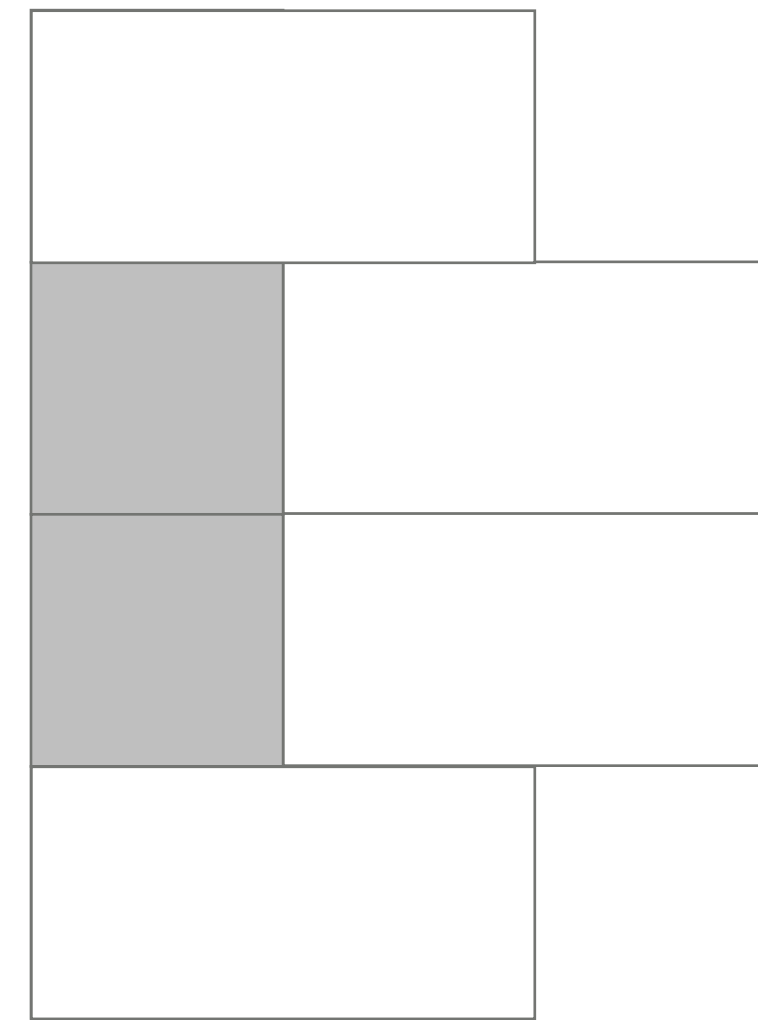
14

<https://www.acmicpc.net/problem/2718>

- $G[N] = F[N-1] + G[N-2]$



$F[N-1]$



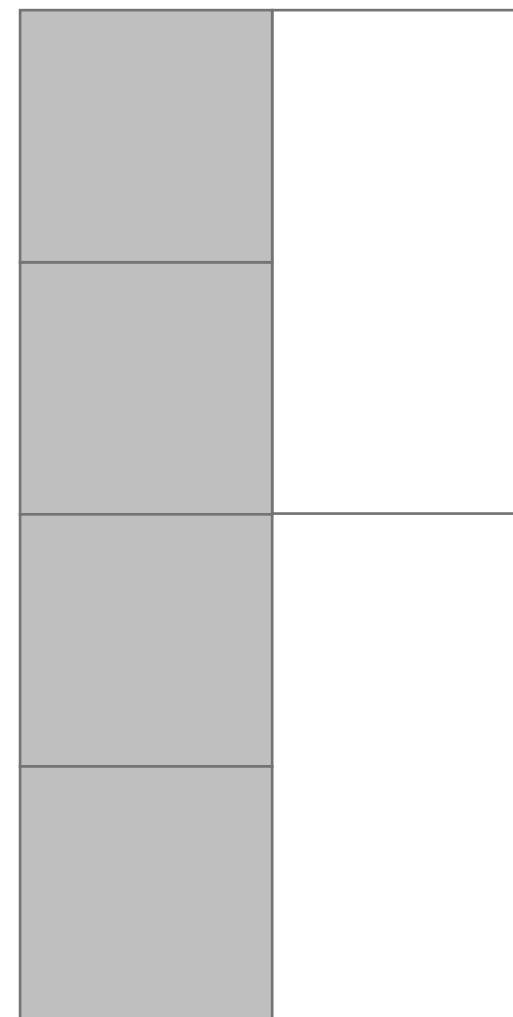
$G[N-2]$

타일 채우기

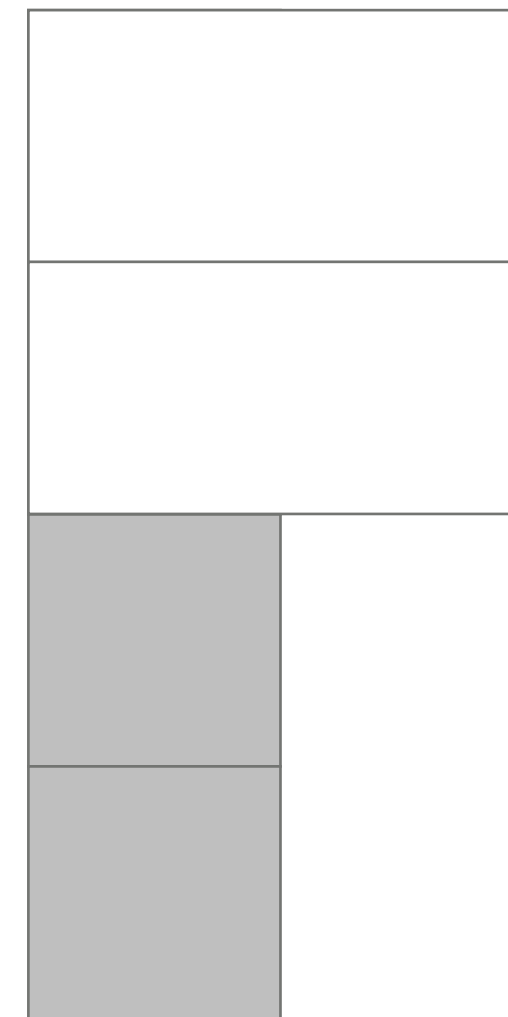
15

<https://www.acmicpc.net/problem/2718>

- $H[N] = F[N-1] + H[N-1]$



$F[N-1]$



$H[N-1]$

타일 채우기

<https://www.acmicpc.net/problem/2718>

- $F[0] = 1$
- $F[1] = 1$
- $G[0] = 0$
- $G[1] = 1$
- $H[0] = 0$
- $H[1] = 1$

타일 채우기

17

<https://www.acmicpc.net/problem/2718>

- 소스: <http://codeplus.codes/6953d6a2a46e49c281bdb536e9f59e2a>

체스로 도미노를 타자

<https://www.acmicpc.net/problem/9936>

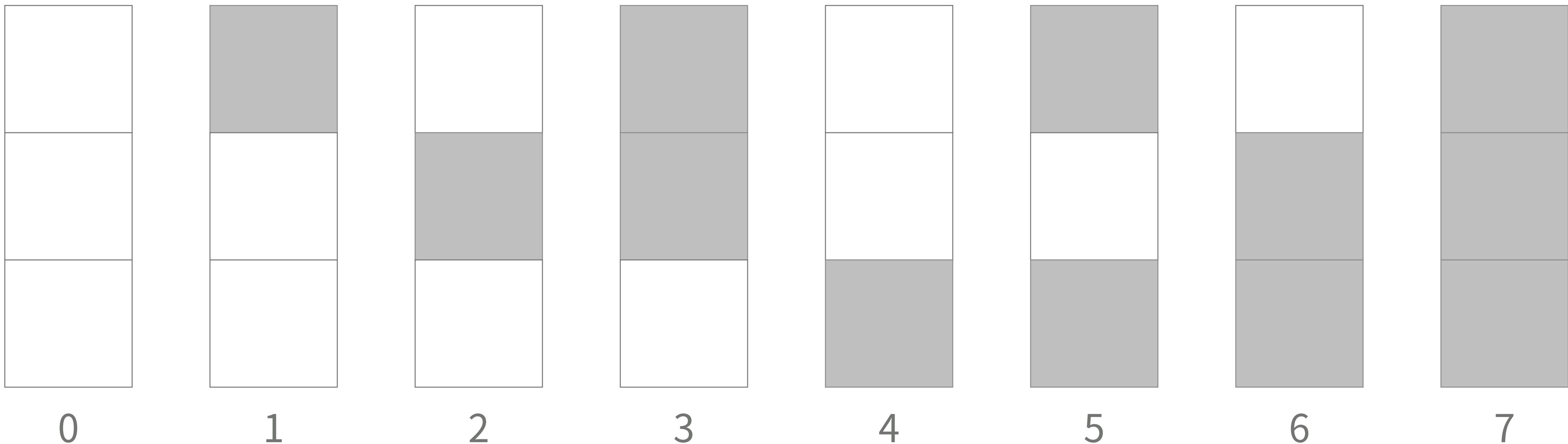
- N행 3열로 이루어진 체스판이 있다. 체스판에는 정수가 적혀있다. ($1 \leq N \leq 1,000$)
- 크기가 2×1 인 도미노 K개를 이용해 체스판을 덮으려고 한다. ($1 \leq K \leq 1,000$)
- 덮인 칸의 합의 최댓값을 구하는 문제

체스로 도미노를 타자

19

<https://www.acmicpc.net/problem/9936>

- $D[i][j][k] = 3 \times i$ 를, i 열의 상태는 k, j 개 도미노를 놓았을 때 최대 점수
- 마지막에 올 수 있는 가능한 경우의 수 (회색: 채워져 있는 칸)



체스로 도미노를 타자

20

<https://www.acmicpc.net/problem/9936>

- 소스: <http://codeplus.codes/1e5a0a4e1781415190ceb9f383d8c7ae>

<https://www.acmicpc.net/problem/1086>

- 서로 다른 정수 N 개로 이루어진 집합이 있다 $1 \leq N \leq 15$, $1 \leq$ 정수의 길이 ≤ 15
- 이 집합의 순열을 합치면 큰 정수 하나를 만들 수 있다
- 예를 들어, $\{5221, 40, 1, 58, 9\}$ 로 5221401589를 만들 수 있다
- 합친수가 정수 K 로 나누어 떨어지는 순열을 구하는 프로그램을 작성하시오 $1 \leq K \leq 100$
- 그냥 랜덤하게 순열 하나를 정답이라고 출력하려고 한다
- 이 문제에는 정답이 여러 개 있을 수도 있고, 우연히 문제의 정답을 맞출 수도 있다.
- 우연히 정답을 맞출 확률을 분수로 출력하는 프로그램을 작성하시오

<https://www.acmicpc.net/problem/1086>

- $D[S][M]$ = 사용한 순열에 포함된 수의 집합이 S이고, 나머지가 M인 것의 개수
- S에 포함되어 있지 않은 수를 L이라고 했을 때
- L번째 수 : $A[L]$
- L번째 수의 길이: $Len[L]$
- L번째 수가 포함된 경우의 나머지 = $(M * 10^{Len[L]} + A[L]) \% M$
- $D[S \mid (1 \ll L)][next] += D[S][M]$

박성원

23

<https://www.acmicpc.net/problem/1086>

- 소스: <http://codeplus.codes/6a3fe57e5e8d4c9e824646c6d51f083f>

소수 만들기

<https://www.acmicpc.net/problem/1555>

- 1보다 크거나 같고, 30보다 작거나 같은 수로 채워진 크기 N 의 배열 A ($1 \leq N \leq 6$)
- 이 배열의 원소를 $+$, $-$, $*$, $/$, 괄호를 원하는 만큼 써서 식을 만들어서, 가장 큰 소수와 가장 작은 소수를 구하는 문제

소수 만들기

25

<https://www.acmicpc.net/problem/1555>

- $D[S]$ = 사용한 소수의 상태가 S 일 때, 만들 수 있는 모든 소수

소수 만들기

<https://www.acmicpc.net/problem/1555>

- S를 first와 second로 나눈 다음에, 각각 문제를 풀어서
- D[first]와 D[second]를 구한다
- 그 다음, D[first]에 들어있는 모든 x와 D[second]에 들어있는 모든 y에 대해서
- 만들 수 있는 모든 수를 만들어본다.

소수 만들기

<https://www.acmicpc.net/problem/1555>

- x/y
- y/x
- $x+y$
- $x-y$
- $y-x$
- $x*y$

소수 만들기

<https://www.acmicpc.net/problem/1555>

- 소스: <http://codeplus.codes/a2968574313540e7ab47fb7e2154207d>

네 부분문자열

<https://www.acmicpc.net/problem/1479>

- 문자열 S와 네 부분문자열 a, b, c, d가 주어졌을 때
- 각각의 부분문자열 a, b, c, d가 S와 어디에서 겹쳐졌는지 구해보자
- 이 때, S의 어떤 위치에 있는 문자가 a, b, c, d중 하나에 속했을 때, 이 문자를 가릴 수 있다고 한다.
- 가려진 문자의 최소값과 최대값을 구하는 문제
- $1 \leq S$ 의 길이 ≤ 2500 , $1 \leq a, b, c, d$ 의 길이 ≤ 50
- $S = abacaba$, $a = ab$, $b = ba$, $c = a$, $d = c$ 인 경우 정답: 4, 6

네 부분문자열

<https://www.acmicpc.net/problem/1479>

f	o	u	r	s	u	b	s	t	r	i	n	g	s
	o	u	r										
				s									
									r	i	n	g	
				s	u	b							
	+	+	+	+	+	+			+	+	+	+	

네 부분문자열

<https://www.acmicpc.net/problem/1479>

- 왼쪽에서 부터 한 글자씩 보면서 그 위치부터 부분 문자열로 가릴 수 있는지 확인한다
- 얼마나 많은 글자를 이미 가렸는지
- 그리고, 어떤 글자를 사용했는지 알아내는 것이 필요하다

네 부분문자열

<https://www.acmicpc.net/problem/1479>

- $D[index][covered][mask]$
- 현재 index 번째를 보고 있고, 가려진 문자의 수는 covered개이고, 사용한 부분 문자열은 mask 이다
- $D[1][3][0001_2]$

f	o	u	r	s	u	b	s	t	r	i	n	g	s
---	---	---	---	---	---	---	---	---	---	---	---	---	---

네 부분문자열

<https://www.acmicpc.net/problem/1479>

```
d[index][covered][mask] = d[index+1][max(0, covered-1)][mask];
for (int i=0; i<4; i++) {
    if ((mask&(1<<i)) != 0 || !starts[index][i]) continue;
    int nextc = max(covered, len[i]);
    int nextm = mask | (1<<i);
    calc(index, nextc, nextm);
    d[index][covered][mask] = min(d[index][covered][mask],
                                   d[index][nextc][nextm] + max(0, len[i]-covered));
}
```

네 부분문자열

<https://www.acmicpc.net/problem/1479>

- 소스: <http://codeplus.codes/4d0f447512894975bbfda409f418fe34>

두부장수 장홍준

<https://www.acmicpc.net/problem/1657>

- 세로크기 N , 가로크기 M 인 두부판을 가지고 2×1 짜리 두부로 잘라서 판다.
- 두부판의 위치마다 등급이 다르다.
- $1 \leq N, M \leq 14$

	A	B	C	D	F
A	10	8	7	5	1
B	8	6	4	3	1
C	7	4	3	2	1
D	5	3	2	2	1
F	1	1	1	1	0

두부장수 장홍준

<https://www.acmicpc.net/problem/1657>

- 격자판 채우기와 비슷하게 해결한다

두부장수 장홍준

37

<https://www.acmicpc.net/problem/1657>

- 소스: <http://codeplus.codes/5802e599d6754ceeb5214349594ce741>

행렬과 다이나믹

동민 수열

<https://www.acmicpc.net/problem/1529>

- 4와 7로 이루어진 수를 금민수라고 한다
- 수 N개가 주어졌을 때, 만들 수 있는 길이가 L인 동민 수열의 개수를 구하는 문제 ($1 \leq L \leq 10^9$)
- 다음 조건을 만족하는 수열 A를 동민 수열이라고 한다
 1. 수열을 구성하는 수가 금민수이다
 2. 수열을 구성하는 수가 입력으로 주어지는 수에 있어야 한다
 3. 모든 i에 대해서, A[i]의 마지막 자리는 A[i+1]의 첫 번째 자리와 같아야 한다

동민 수열

<https://www.acmicpc.net/problem/1529>

- 4와 7로 이루어진 수를 금민수라고 한다
- 수 N개가 주어졌을 때, 만들 수 있는 길이가 L인 동민 수열의 개수를 구하는 문제 ($1 \leq L \leq 10^9$)
- 다음 조건을 만족하는 수열 A를 동민 수열이라고 한다
 1. 수열을 구성하는 수가 금민수이다
 2. 수열을 구성하는 수가 입력으로 주어지는 수에 있어야 한다
 3. 모든 i에 대해서, A[i]의 마지막 자리는 A[i+1]의 첫 번째 자리와 같아야 한다

동민 수열

<https://www.acmicpc.net/problem/1529>

- 4와 7로 이루어진 수를 금민수라고 한다
- 수 N개가 주어졌을 때, 만들 수 있는 길이가 L인 동민 수열의 개수를 구하는 문제 ($1 \leq L \leq 10^9$)
- 다음 조건을 만족하는 수열 A를 동민 수열이라고 한다
 1. 수열을 구성하는 수가 금민수이다
 2. 수열을 구성하는 수가 입력으로 주어지는 수에 있어야 한다
 3. 모든 i에 대해서, A[i]의 마지막 자리는 A[i+1]의 첫 번째 자리와 같아야 한다
- 실제 수가 무엇인지 중요한 것이 아니고, 첫 숫자와 마지막 숫자가 중요하다

동민 수열

<https://www.acmicpc.net/problem/1529>

- 4로 시작하고 4로 끝나는 수의 개수를 c44
- 4로 시작하고 7로 끝나는 수의 개수를 c47
- 7로 시작하고 4로 끝나는 수의 개수를 c74
- 7로 시작하고 7로 끝나는 수의 개수를 c77

동민 수열

<https://www.acmicpc.net/problem/1529>

- 4로 시작하고 4로 끝나는 길이가 n 인 동민 수열의 개수를 $d44[n]$
- 4로 시작하고 7로 끝나는 길이가 n 인 동민 수열의 개수를 $d47[n]$
- 7로 시작하고 4로 끝나는 길이가 n 인 동민 수열의 개수를 $d74[n]$
- 7로 시작하고 7로 끝나는 길이가 n 인 동민 수열의 개수를 $d77[n]$

동민 수열

<https://www.acmicpc.net/problem/1529>

- 4로 시작하고 4로 끝나는 길이가 n 인 동민 수열의 개수를 $d44[n]$
 - 4로 시작하고 7로 끝나는 길이가 n 인 동민 수열의 개수를 $d47[n]$
 - 7로 시작하고 4로 끝나는 길이가 n 인 동민 수열의 개수를 $d74[n]$
 - 7로 시작하고 7로 끝나는 길이가 n 인 동민 수열의 개수를 $d77[n]$
-
- $d47[n] = d44[n-1] * c47 + d47[n] * c77$

동민 수열

<https://www.acmicpc.net/problem/1529>

- 그런데, 수열의 길이가 너무 길다.

동민 수열

<https://www.acmicpc.net/problem/1529>

- 그런데, 수열의 길이가 너무 길다.
- 다음과 같은 행렬을 만들어서 수열의 개수를 구할 수 있다.
- $A = \begin{pmatrix} c_{44} & c_{47} \\ c_{74} & c_{77} \end{pmatrix}$
- 정답은?

동민 수열

47

<https://www.acmicpc.net/problem/1529>

- 그런데, 수열의 길이가 너무 길다.
- 다음과 같은 행렬을 만들어서 수열의 개수를 구할 수 있다.
- $A = \begin{pmatrix} c_{44} & c_{47} \\ c_{74} & c_{77} \end{pmatrix}$
- 정답은?
- A^n 의 모든 값을 더한다

동민 수열

<https://www.acmicpc.net/problem/1529>

- 소스: <http://codeplus.codes/c0bc58433e834d8e8db6eeca999098b7>

체스판 이동

<https://www.acmicpc.net/problem/18287>

- 크기가 $N \times M$ 인 체스판이 있다. $1 \leq N \leq 10^9, 1 \leq M \leq 30$
- (i, j) 의 색상은 $(i+j)$ 가 2로 나누어 떨어지면 검정색, 아니면 흰색
- 이동은 행 번호가 증가하는 쪽으로만 이동이 가능
- 변이나 점을 공유하면 인접한 칸이다.
- 행 번호가 홀수이면, 색이 같은 인접한 칸으로, 짝수이면 아무 인접한 칸으로 이동 가능
- 1번 행에서 이동을 시작했을 때, N번 행에 도착하는 방법의 수를 구하는 문제

체스판 이동

50

<https://www.acmicpc.net/problem/18287>

- $D[i][j]$ = (i, j)에 도착하는 방법의 수
- $D[i][j] = D[i-1][j-1] + D[i-1][j] + D[i-1][j+1]$ (i가 홀수인 경우)
- $D[i][j] = D[i-1][j-1] + D[i-1][j+1]$ (i가 짝수인 경우)

체스판 이동

<https://www.acmicpc.net/problem/18287>

- N 제한이 10^9 라서 너무 크기 때문에, 행렬을 만들어서 해결하면 된다.

체스판 이동

<https://www.acmicpc.net/problem/18287>

- $D[i][j]$ = (i, j) 에 도착하는 방법의 수
 - $D[i][j] = D[i-1][j-1] + D[i-1][j] + D[i-1][j+1]$ (i 가 홀수인 경우)
 - $D[i][j] = D[i-1][j-1] + D[i-1][j+1]$ (i 가 짝수인 경우)
-
- $D[i] = A \times D[i-1]$ 에서 행렬 A 를 만들어야 하는데, i 가 홀수인 경우와 짝수인 경우 행렬이 다르다.

체스판 이동

<https://www.acmicpc.net/problem/18287>

- $D[i][j]$ = (i, j) 에 도착하는 방법의 수
- $D[i][j] = D[i-1][j-1] + D[i-1][j] + D[i-1][j+1]$ (i 가 홀수인 경우)
- $D[i][j] = D[i-1][j-1] + D[i-1][j+1]$ (i 가 짝수인 경우)
- $D[i] = \text{even} \times D[i-1]$ (i 가 짝수인 경우)
- $D[i] = \text{odd} \times D[i-1]$ (i 가 홀수인 경우)
- 홀수와 짝수는 번갈아가면서 나오기 때문에
- $D[i] = (\text{even} \times \text{odd}) \times D[i-2]$ 라고 볼 수 있다.
- $A = \text{even} \times \text{odd}$ 라고 했을 때, $D[i] = A \times D[i-2]$ 로 계산할 수 있다.

체스판 이동

<https://www.acmicpc.net/problem/18287>

- 소스: <http://codeplus.codes/92606abca78e4b0d9595b726e81e8fe7>

팀 연습

<https://www.acmicpc.net/problem/18288>

- A, B, C 세 사람이 N개의 문제를 풀려고 한다. $1 \leq N \leq 10^5$
- 문제는 1번부터 N번까지 순서대로 풀어야 한다.
- 각 문제는 한 사람만 해결할 수 있다.
- A가 해결한 문제의 수는 K의 배수가 되어야 한다. $0 \leq K \leq 10$
- B는 문제를 연속해서 풀 수 없다.
- C는 한 문제 이상 해결해야 한다.

팀 연습

<https://www.acmicpc.net/problem/18288>

- $D[N][A][B][C]$
 - N: 풀어야 하는 문제의 수
 - A: A가 푼 문제의 수를 K로 나눈 나머지
 - B: N-1번 문제를 B가 풀었으면 1, 아니면 0
 - C: C가 푼 문제의 수가 한 문제 이상이면 1, 아니면 0

팀 연습

<https://www.acmicpc.net/problem/18288>

```
long long go(int index, int a, int b, int c) {
    if (index == n) {
        if (a == 0 && c == 1) return 1; else return 0;
    }
    long long &ans = d[index][a][b][c];
    if (ans != -1) return ans;
    ans = 0;
    if (k > 0) ans += go(index+1, (a+1)%k, 0, c);
    if (b == 0) ans += go(index+1, a, 1, c);
    ans += go(index+1, a, 0, 1);
    ans %= mod;
    return ans;
}
```

팀 연습

58

<https://www.acmicpc.net/problem/18288>

- 소스: <http://codeplus.codes/41f4c659b52243e4b70e2b9e871be8db>

팀 연습 더

<https://www.acmicpc.net/problem/18289>

- A, B, C 세 사람이 N개의 문제를 풀려고 한다. $1 \leq N \leq 10^{18}$
- 문제는 1번부터 N번까지 순서대로 풀어야 한다.
- 각 문제는 한 사람만 해결할 수 있다.
- A가 해결한 문제의 수는 K의 배수가 되어야 한다. $0 \leq K \leq 10$
- B는 문제를 연속해서 풀 수 없다.
- C는 한 문제 이상 해결해야 한다.

팀 연습 더

<https://www.acmicpc.net/problem/18289>

- $D[N][A][B][C]$
 - N: 풀어야 하는 문제의 수
 - A: A가 푼 문제의 수를 K로 나눈 나머지
 - B: N-1번 문제를 B가 풀었으면 1, 아니면 0
 - C: C가 푼 문제의 수가 한 문제 이상이면 1, 아니면 0
- $D[N] = A \times D[N-1]$ 와 같은 행렬의 곱으로 나타내야 하는데, 뒤 배열이 3차원 배열이다.

팀 연습 더

<https://www.acmicpc.net/problem/18289>

- $D[N][A][B][C]$
 - N: 풀어야 하는 문제의 수
 - A: A가 푼 문제의 수를 K로 나눈 나머지
 - B: N-1번 문제를 B가 풀었으면 1, 아니면 0
 - C: C가 푼 문제의 수가 한 문제 이상이면 1, 아니면 0
- $D[N] = A \times D[N-1]$ 와 같은 행렬의 곱으로 나타내야 하는데, 뒤 배열이 3차원 배열이다.
- $[A][B][C]$ 를 1차원 배열 크기가 $K \times 2 \times 2$ 인 [state]로 나타낼 수 있다.
- 그럼 행렬 곱을 사용할 수 있다.

팀 연습 더

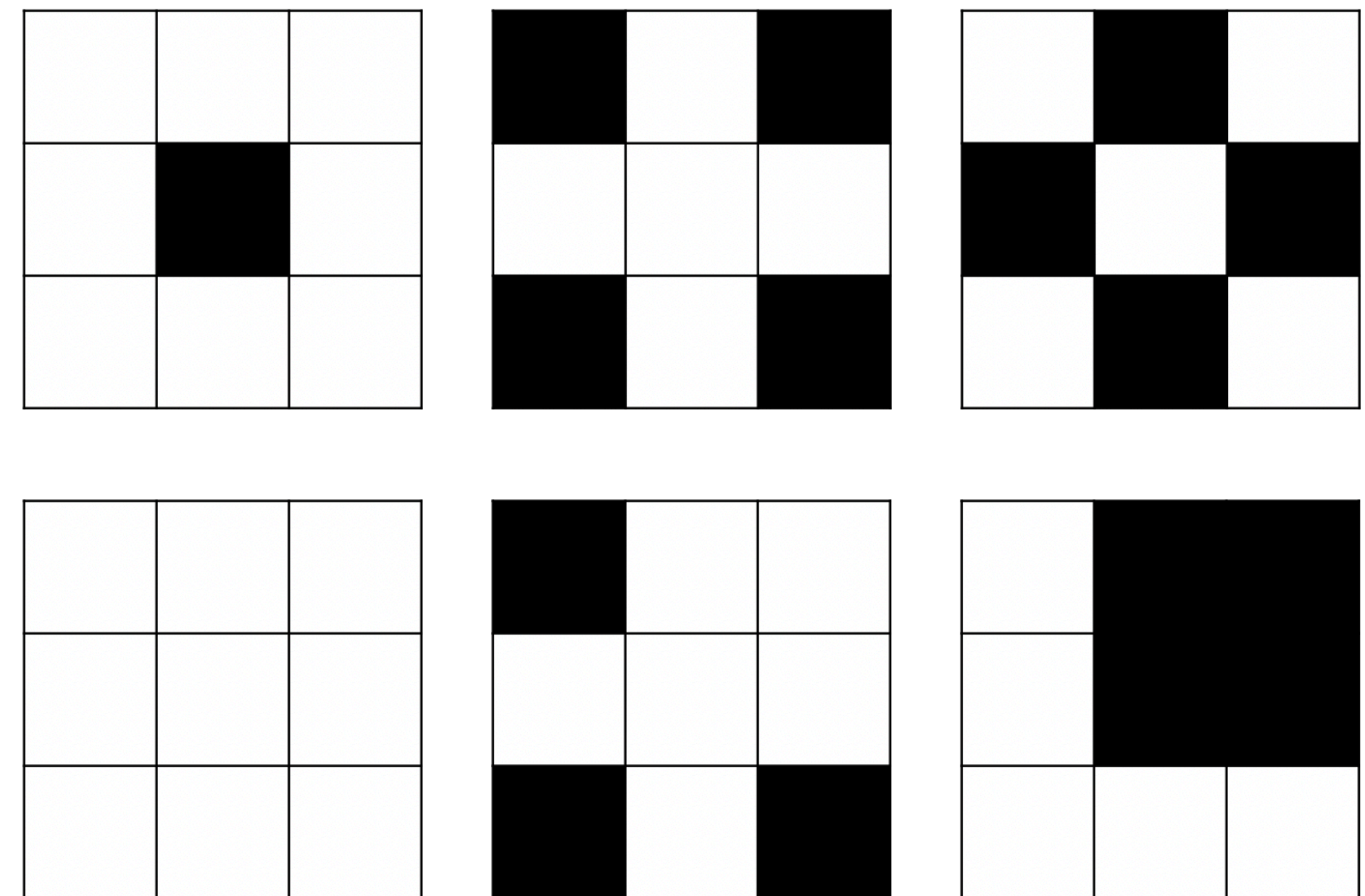
<https://www.acmicpc.net/problem/18289>

- 소스: <http://codeplus.codes/4df98e8a718a46768aa5a40c5132d714>

직사각형 색칠 2

<https://www.acmicpc.net/problem/18286>

- $N \times M$ 크기의 직사각형이 있을 때, 각 칸을 검정색 또는 흰색으로 색칠해야 한다.
- 같은 색으로 이루어진 2×2 크기의 정사각형이 없어야 할 때 방법의 수
- $1 \leq N \leq 10^{18}, 1 \leq M \leq 5$



직사각형 색칠 2

<https://www.acmicpc.net/problem/18286>

- $D[\text{row}][\text{state}]$ = row행의 색칠된 상태가 state일 때 방법의 수
- $d[\text{row}][\text{state}] = \sum d[\text{row}-1][\text{pstate}]$
 - row-1행의 상태는 pstate이고, row행의 상태는 state, 여기에 2×2 크기의 같은 색이 없음

직사각형 색칠 2

<https://www.acmicpc.net/problem/18286>

```
for (int i=0; i<(1<<m); i++) {
    d[1][i] = 1;
}
for (int i=2; i<=n; i++) {
    for (int j=0; j<(1<<m); j++) {
        for (int k=0; k<(1<<m); k++) {
            if (ok(k, j)) {
                d[i][j] += d[i-1][k];
                d[i][j] %= mod;
            }
        }
    }
}
```

직사각형 색칠 2

<https://www.acmicpc.net/problem/18286>

```
int isset(int state, int index) {
    if (state & (1 << index)) return 1;
    else return 0;
}

bool ok(int pstate, int state) {
    for (int i=0; i<m-1; i++) {
        int color11 = isset(pstate, i), color12 = isset(pstate, i+1);
        int color21 = isset(state, i), color22 = isset(state, i+1);
        if (color11 == color12 && color12 == color21 && color21 ==
color22) return false;
    }
    return true;
}
```

직사각형 색칠 2

<https://www.acmicpc.net/problem/18286>

- 하지만 행의 개수 제한이 10^{18} 이다.

직사각형 색칠 2

<https://www.acmicpc.net/problem/18286>

- 크기가 $2^5 \times 2^5$ 인 행렬을 만들 수 있다.
- 이 행렬을 N제공하면 정답을 구할 수 있다.

직사각형 색칠 2

69

<https://www.acmicpc.net/problem/18286>

- 소스: <http://codeplus.codes/455d65e06a0d47e08bcf8229cfff4d9f>

나이트

<https://www.acmicpc.net/problem/10562>

- M행 N열의 체스판에 나이트를 놓는 경우의 수를 구하는 문제
- $1 \leq M \leq 4, 1 \leq N \leq 10^9$

나이트

<https://www.acmicpc.net/problem/10562>

- 나이트가 공격할 수 있는 열의 범위는 2칸이기 때문에,
- $D[N][S]$ = N열까지 채웠고, 마지막 두 열의 상태가 S일 때, 경우의 수

나이트

<https://www.acmicpc.net/problem/10562>

- 상태의 개수가 2^8 가지 이기 때문에, 총 256×256 크기의 행렬이 필요하다.
- 코드를 이용해서 다이나믹 점화식을 세워야 한다

나이트

<https://www.acmicpc.net/problem/10562>

- 소스: <http://codeplus.codes/d5f7d713fd78462e95f9d8df59f2fc0e>