SW 역량 테스트 대비반 – 5회차

숙명여자대학교

2020년 08월 12일

Mid-Test (문제 1)

아이
디어

1. 숫자, 스트라이크, 볼을 저장할 자료구조로 2차원 벡터를 활용
2. 조건이 3자리의 서로 다른 자연수이므로 123 -> 987까지 스트라이크, 볼 여부를 체크
3. 문자로 바꿔서 한자리씩 스트라이크 볼 여부를 체크
4. 주어진 조건과 모두 일치하면 카운트 및 해당 숫자를 저장 후, 출력

Mid-Test (문제 2)

아이디어

1. 작업의 우선순위를 두는 것이 중요
2. 입력은 Si, Ei, Wi를 받기 위한 구조체 배열로 선언
3. 시작 날짜가 빠르고 시작 날짜가 같으면 종료 날짜가 빠른 작업을 우선으로 하여 정렬 수행
4. 정렬된 작업에 따라 최대 8시간 작업을 하도록 배열의 Wi를 줄여주고.. (작업이 끝나지 않았다면)
5. 작업이 끝났다면 당일날 작업을 다음날로 넘겨준다.
6. 3~5를 반복
7. 작업이 모두 끝났다면 YES, 그렇지 않다면 NO를 출력

Mid-Test

1번 : https://onlinegdb.com/S1hMZvUo8

2번 (C++) : https://onlinegdb.com/rkMEWw8s8

2번 (C) : https://onlinegdb.com/rJEr-wUiL

오늘 할 내용

수학적 귀납법 (분할과 정복)

구분	
1주차 (07/08)	
2주차 (07/15)	
3주차 (07/22)	
4주차 (07/29)	
5주차 (08/05)	
6주차 (08/12)	분할과 정복 강의 및 실습
7주차 (08/19)	동적 계획법 강의 및 실습
8주차 (08/26)	그래프 알고리즘 강의 및 실습
(옵션) 9주차 ~	실전 감각 기르기 (실전 기출 기반 문제풀이 집중반)

수학적 귀납법

수학적 귀납법은 자연수 n에 관한 명제 P(n)이 모든 자연수에 대해서 성립함을 증명하기 위한 수학의 증명법 중 한 방법이다.

자연수 n에 관한 명제 $P(n)$이 모든 자연수 n에 대해 성립함을 다음과 같은 2가지 단계로 증명한다.

① $P(1)$이 성립함을 보인다 – Basis
② $P(k)$가 성립한다고 가정하고 $P(k+1)$이 성립함을 보인다 – induction

①, ②를 통해서 모든 자연수에 대해서 $P(n)$이 성립함을 보이는 방법이다.

수학적 귀납법을 이용한 문제해결의 절차는 다음과 같다.

① 입력값이 n인 문제의 해를 $f(n)$으로 정의한다.
② $f(1)$을 직접 구하여 출력한다.
③ $f(k)$를 이미 구해두었다고 가정하고 $f(k)$를 통하여 $f(n)$을 구하고 출력한다.

여기서 k는 $n-1$이 될 수도 있고, $\frac{n}{2}$이 될 수도 있다. 여기서 중요한 것은 k를 문제에 따라 적절한 형태로 설계해야 한다는 점이다.

문제풀이 - 수학적 귀납법

임의의 정수 n을 입력 받아 1부터 n까지 합을 구하는 프로그램을 작성하시오.
(단, $n \le 100$) (재귀함수 이용, 수학적 귀납법을 이용)

INPUT	OUTPUT
10	55
100	5050

답안

일단 먼저 함수 $f(n)$을 다음과 같이 정의하자.

$$f(n) = \text{'1부터 } n\text{까지의 합'}$$

다음으로 $f(1)$을 정의에 의하여 직접 구하면

$$f(1) = 1$$

$f(n-1)$을 이미 구해두었다고 가정하면, 다음과 같이 $f(n)$을 구할 수 있다. $f(10)$을 $f(9)$를 이용하여 나타내보자.

$$f(10) = 1+2+3+4+5+6+7+8+9+10$$
$$f(9) = 1+2+3+4+5+6+7+8+9$$

따라서 $f(10) = f(9) + 10$ 과 같다. 이를 일반화하면,

$$f(n) = f(n-1) + n$$

```
#include <iostream>
using namespace std;

int f(int n)
{
  if(n==1) return 1;
  return f(n-1)+n;
}

int main()
{
    int n;
    cin >> n;
    cout << f(n);
  return 0;
}
```

문제풀이 - 수학적 귀납법 (k진법)

임의의 10진법 정수 n을 입력받아 이를 k진법으로 출력하는 프로그램을 작성하시오.
(단, $n \leq 100{,}000$ $k \leq 10$) (수학적 귀납법을 이용)

INPUT	OUTPUT
12 2	1100
100 8	144

문제풀이 - 수학적 귀납법 (k진법)

아이디어

12
2
6 ··· 0 ④
2
3 ··· 0 ③
2
1 ··· 1 ②
①

마지막의 몫이 2보다 작으므로 더 이상 할 필요가 없다.

이 때 맨 아래 2보다 작은 1인 몫과 나머지를 아래에서부터 위로 하나씩 ①로부터 ④까지 연결한

1100

이라는 값이 12를 이진법으로 바꾼 값이 된다.

$$f(n,\ k) = \begin{cases} n & (n < k) \\ f(n/k,\ k),\ print(n\%k) & (n \geq k) \end{cases}$$

문제풀이 - 수학적 귀납법 (k진법)

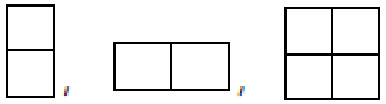
답안

https://onlinegdb.com/r1Ih7wIi8

수학적 귀납법 실습 (타일채우기)

타일채우기(S)

2*1 혹은 2*2크기의 타일을 2*n 크기의 직사각형모양 틀에 넣으려고 한다. 이 때 가능한 경우의 수를 구하여라.



경우의 수가 커지므로, 주어지는 수 m으로 나눈 나머지를 출력한다.

입력

첫 줄에는 직사각형 틀의 가로 길이 n이 주어진다.

둘째 줄에는 m이 주어진다. (1<=n<=100,000, 1<=m<=40,000)

출력

경우의 수를 m으로 나눈 나머지를 출력한다.

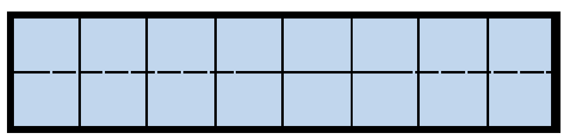
입력 예	출력 예
8 100	71
8 2	1

수학적 귀납법 실습 (타일채우기)

먼저 첫 번째 방법을 알아보자. 문제에서 구하고자 하는 해는 $f(n)$이므로 먼저 $f(n)$을 구하기 위해 $f(1)$, $f(2)$, $\cdots$, $f(n-1)$은 이미 구해두었다고 가정한다.

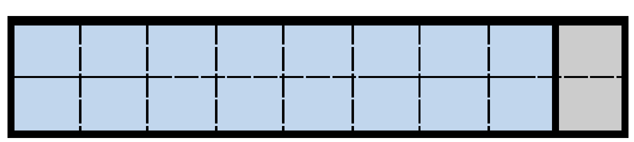
어떤 $f(k)$로부터 $f(n)$을 만드는 데 마지막으로 사용할 수 있는 타일은 3가지가 있다. 먼저 2×1타일을 이용하여 $f(n)$으로 만드는 경우를 살펴보자.

이 경우 $f(n-1)$까지는 모두 채워져 있어야 2×1 타일을 이용하여 $f(n)$을 만들 수 있다.



$n-1$

위 그림과 같이 $n-1$칸을 채운 경우의 수를 $f(n-1)$이라고 알고 있다고 가정하자. 이 상태에서 $f(n)$을 구하기 위하여 2×1 의 타일을 이용할 수 있다.



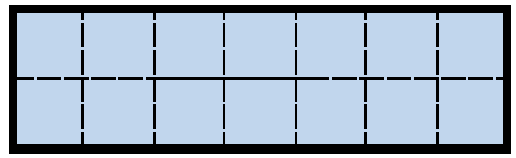
n

이와 같이 $f(n)$을 만드는 경우의 수는 얼마일까?

위의 그림과 같이 $n-1$가지를 채운 경우로부터 2×1 의 타일을 이용하여 $f(n)$을 만들 수 있는 경우는 그냥 뒤에 붙이는 1가지뿐이다. 따라서 $f(n)$ 중 마지막에 도미노를 세우는 방법은 $f(n-1)$이라는 사실을 알 수 있다.

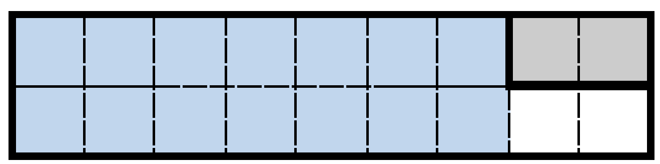
즉 2×1 타일을 이용하여 $f(n)$을 만들 경우의 수는 $f(n-1)$과 같다. 즉 $f(n-1)$을 모두 채우는 경우의 수를 알면 $f(n)$중 2×1을 이용하여 채우는 경우의 수를 알 수 있다.

수학적 귀납법 실습 (타일채우기)



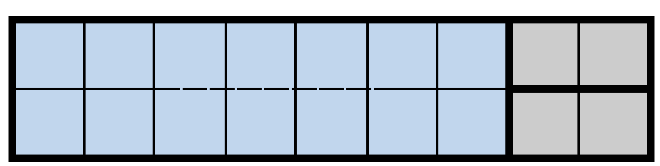
$n-2$

위 그림과 같이 $n-2$칸을 채운 경우의 수를 $f(n-2)$라고 알고 있다고 가정하자. 이 상태에서 $f(n)$을 구하기 위하여 1×2의 타일을 이용할 수 있다. 먼저 1개를 배치해보면 다음과 같다.



n

이 때 마지막 부분이 비어있게 된다. 이 부분을 채울 수 있는 방법은 다시 1×2타일을 하나 더 쓰는 수밖에 없다. 따라서 마지막을 1×2타일로 채우는 방법은 다음 그림과 같다.

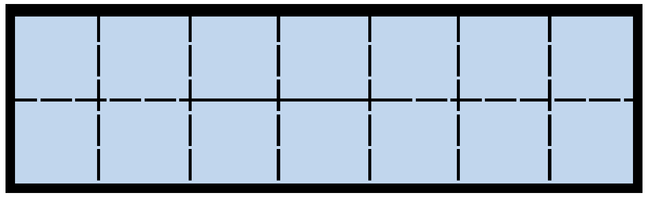


n

위와 같은 방법 1가지뿐이다. 따라서 마지막에 1×2타일을 이용하여 $f(n)$을 만들 수 있는 방법은 $f(n-2)$가지임을 알 수 있다.

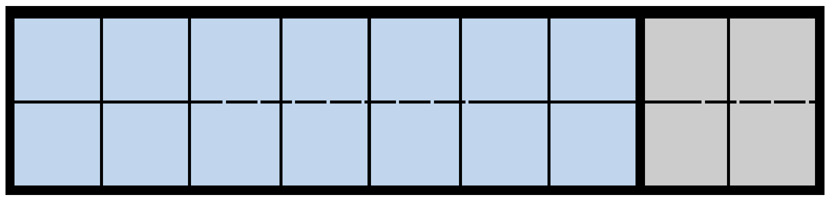
$f(n)$을 만들기 위해서 마지막으로 1×2 타일을 이용하는 경우의 수는 $f(n-2)$가지임을 알 수 있다.

수학적 귀납법 실습 (타일채우기)



$n-2$

위 그림과 같이 $n-2$칸을 채운 경우의 수를 $f(n-2)$라고 알고 있다고 가정하자. 이 상태에서 $f(n)$을 구하기 위하여 2×2의 타일을 이용할 수 있다. 먼저 1개를 배치해보면 다음과 같다.



n

이와 같이 $f(n)$을 만들 수 있으며, 위와 같은 방법 1가지뿐이다. 따라서 마지막에 2×2타일을 이용하여 $f(n)$을 만들 수 있는 방법은 $f(n-2)$가지임을 알 수 있다.

수학적 귀납법 실습 (타일채우기)

따라서 위 관계로부터 각 사건은 합의 법칙이 적용됨을 알 수 있으며, 이를 통하여 얻어낸 관계식 혹은 점화식은 다음과 같다.

$$
\begin{aligned}
f(n) &= f(n-1) + f(n-2) + f(n-2) \\
&= f(n-1) + 2f(n-2)
\end{aligned}
$$

$$
f(n) = \begin{cases} 1 & (n=0) \\ 1 & (n=1) \\ f(n-1) + 2f(n-2) & (n>1) \end{cases}
$$

혹은

$$
f(n) = \begin{cases} 1 & (n=1) \\ 3 & (n=2) \\ f(n-1) + 2f(n-2) & (n>2) \end{cases}
$$

위 두 식은 같은 의미로 볼 수 있다. 위 두 가지 관계식 중 첫 번째 관계식은 다음과 같이 고쳐도 무방하다.

$$
f(n) = \begin{cases} 1 & (n \le 1) \\ f(n-1) + 2f(n-2) & (n>1) \end{cases}
$$

수학적 귀납법 실습 (타일채우기)

https://onlinegdb.com/By7TbwLoI

하노이의 탑

https://www.acmicpc.net/problem/11729

문제

세 개의 장대가 있고 첫 번째 장대에는 반경이 서로 다른 n개의 원판이 쌓여 있다. 각 원판은 반경이 큰 순서대로 쌓여있다. 이제 수도승들이 다음 규칙에 따라 첫 번째 장대에서 세 번째 장대로 옮기려 한다.

1. 한 번에 한 개의 원판만을 다른 탑으로 옮길 수 있다.
2. 쌓아 놓은 원판은 항상 위의 것이 아래의 것보다 작아야 한다.

이 작업을 수행하는데 필요한 이동 순서를 출력하는 프로그램을 작성하라. 단, 이동 횟수는 최소가 되어야 한다.

아래 그림은 원판이 5개인 경우의 예시이다.

하노이의 탑

입력

첫째 줄에 첫 번째 장대에 쌓인 원판의 개수 N (1 ≤ N ≤ 20)이 주어진다.

출력

첫째 줄에 옮긴 횟수 K를 출력한다.

두 번째 줄부터 수행 과정을 출력한다. 두 번째 줄부터 K개의 줄에 걸쳐 두 정수 A B를 빈칸을 사이에 두고 출력하는데, 이는 A번째 탑의 가장 위에 있는 원판을 B번째 탑의 가장 위로 옮긴다는 뜻이다.

예제 입력 1 복사

```
3
```

예제 출력 1 복사

```
7
1 3
1 2
3 2
1 3
2 1
2 3
1 3
```

하노이의 탑

아이디어

[풀이] 10원, 50원, 100원, 500원 짜리 동전을 가지고 하노이 탑을 세운다고 생각해 보면 먼저 크기가 가장 작은 3개의 동전을 먼저 A에서 B로 옮겨 놓고 가장 큰 네 번째 동전을 A에서 C로 옮겨 놓은 다음 B에 있던 세 개의 동전을 다시 C에 옮겨 놓는 방법을 생각할 수 있다.

점화식을 만들어 보면 A에 있는 $n-1$개의 고리를 A에서 B로 옮기고, 그리고 가장 큰 고리를 A에서 C로 옮긴 후, B에 있는 $n-1$개의 고리를 C로 옮긴다.

만약 a_n이 n개의 고리를 A에서 C로 옮기는 방법의 수라고 하면,

$$a_n = a_{n-1} + 1 + a_{n-1} = 2a_{n-1} + 1$$

이다. 초기조건은 $a_1 = 1$. 그래서 다음을 얻는다.

$$a_2 = 2a_1 + 1 = 3,$$
$$a_3 = 2a_2 + 1 = 7,$$
$$a_4 = 2a_3 + 1 = 15,$$
$$a_5 = 2a_4 + 1 = 31,$$
$$a_6 = 2a_5 + 1 = 63.$$

이 경우 점화식의 해를 구하는 방법으로 점화식의 해를 추측하고 이를 수학적 귀납법으로 증명하는 것이 있다. 즉 앞의 예에서

$$a_n = 2^n - 1$$

임을 추측할 수 있다. 특별히 이것은 $n=1$에 대해 참이다. 이 해를 a_{n-1}에 대해서 참이라고 가정하면 다음을 얻을 수 있다.

$$a_n = 2a_{n-1} + 1 = 2(2^{n-1} - 1) + 1 = 2^n - 2 + 1 = 2^n - 1$$

만약 "옮긴 횟수"만 출력한다면 점화식은?

f(n) = 1 (n=1)
f(n) = 2f(n-1) + 1 (n>1)

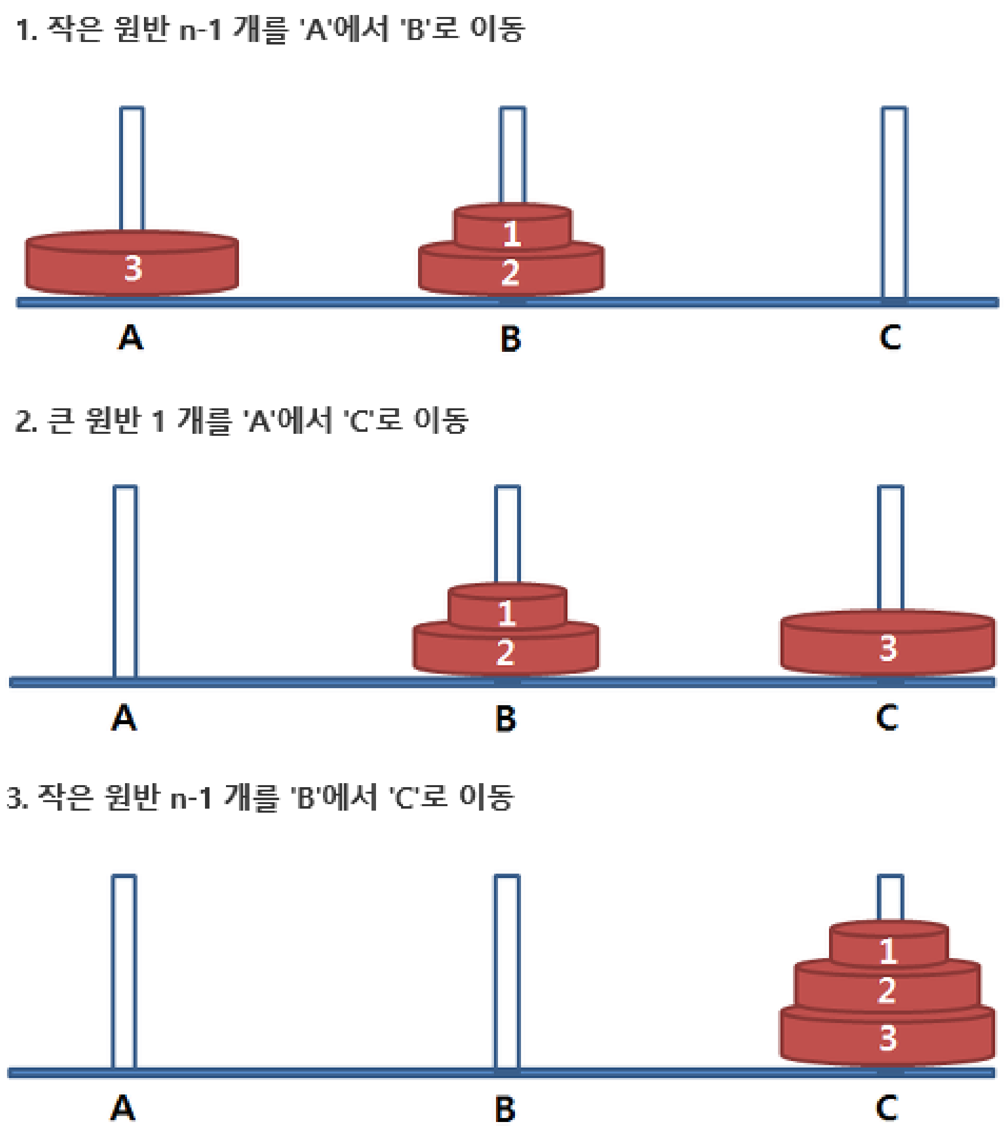
or

f(n) = 2^n - 1 (n>=1)

하노이의 탑

아이디어

일반화 시켜서 점화식을 만들어 보자.



1. 1번 기둥(시작기둥)에 있는 원판중 n-1개를 2번 기둥(중간기둥)으로 옮긴다

2. 1번 기둥에 남아있는1개의 가장 큰 원판을 3번 기둥(도착기둥)으로 옮겨 준다.

3. 아까 1번에서 옮겨놨던 2번 기둥(중간 기둥)에 있던 n-1개의 원판을 다시 3번 기둥으로 옮겨준다

f(n, from, by, to) = push_vector(from, to) (n=1)

f(n, from, by, to) = f(n-1, from, to, by), push_vector(from, to), f(n-1, by, from, to)

하노이의 탑

https://onlinegdb.com/BJXEMwUjL

영역구분

영역 구분

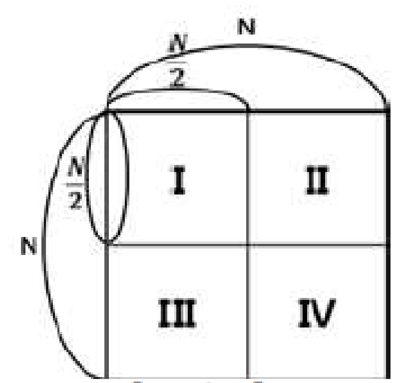
아래 그림과 같이 여러 개의 정사각형 칸들로 이루어진 정사각형 모양의 영역이 주어져 있고, 각 정사각형 칸들은 정올이의 땅은 흰색으로 칠해져 있고 영재의 땅은 검은색으로 칠해져 있다. 주어진 땅을 일정한 규칙에 따라 나누어 다양한 크기를 가진 정사각형 모양의 하얀색 또는 회색 영역으로 구분하려고 한다.

1	1	0	0	0	0	1	1
1	1	0	0	0	0	1	1
0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0
1	0	0	0	1	1	1	1
0	1	0	0	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1

[그림 1]

전체 영역의 크기가 n*n (n= 2^k, k는 1 이상 7 이하의 자연수) 이라면 영역을 구분하는 규칙은 다음과 같다. 전체 영역이 모두 같은 색이 아니라면 가로와 세로로 중간 부분을 잘라서 <그림 2>의 Ⅰ, Ⅱ, Ⅲ, Ⅳ와 같이 똑같은 크기의 네 개의 n/2 * n/2 영역으로 나눈다.

나누어진 영역 Ⅰ, Ⅱ, Ⅲ, Ⅳ 각각에 대해서도 앞에서와 마찬가지로 모두 같은 색으로 이루어지지 않으면 같은 방법으로 똑같은 크기의 네 개의 영역으로 나눈다. 이와 같은 과정을 구분되어진 영역이 모두 하얀색 또는 모두 회색으로 되거나, 하나의 정사각형 칸이 되어 더 이상 나눌 수 없을 때까지 반복한다.

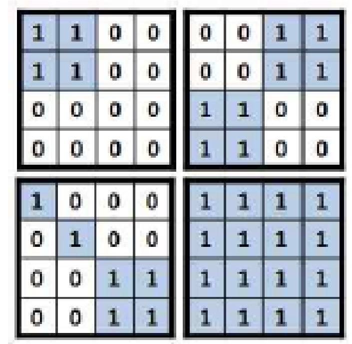


[그림 2]

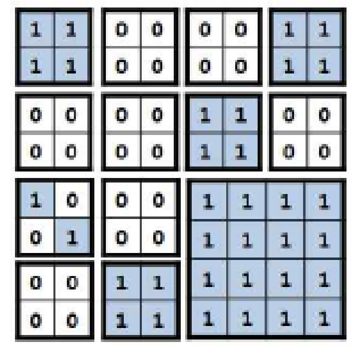
위와 같은 규칙에 따라 나누었을 때 [그림 3]은 [그림 1]의 영역을 처음 나눈 후의 상태를, [그림 4]는 두 번째 나눈 후의 상태를, [그림 5]는 최종적으로 만들어진 다양한 크기의 9개의 하얀색 영역과 7개의 회색영역을 보여주고 있다.

영역구분

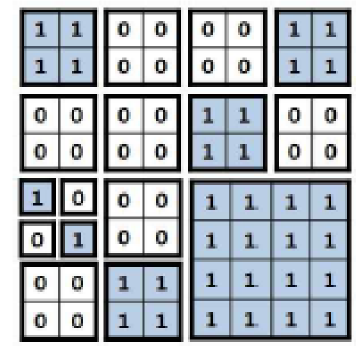
영역 구분 (계속)



[그림 3]



[그림 4]



[그림 5]

입력으로 주어진 영역의 한 변의 길이 n과 각 정사각형 칸의 색 (하얀색 또는 회색)이 주어질 때 잘라진 하얀색 영역과 회색 영역의 개수를 구하는 프로그램을 작성하시오.

입력

입력 파일의 첫째 줄에는 전체 영역의 한 변의 길이 n이 주어져 있다. n은 2, 4, 8, 16, 32, 64, 128 중 하나이다. 영역의 각 가로줄의 정사각형 칸들의 색이 윗 줄부터 차례로 입력 파일의 둘째 줄부터 마지막 줄까지 주어진다. 회색으로 칠해진 칸은 0, 하얀색으로 칠해진 칸은 1로 주어지며 각 숫자 사이에는 빈칸이 하나씩 있다.

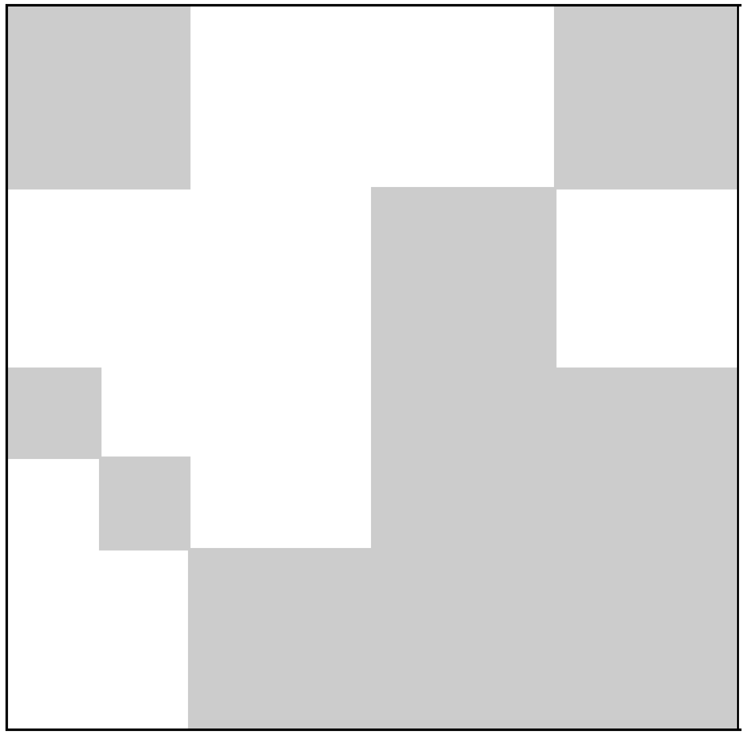
출력

첫째 줄에는 잘라진 하얀색 영역의 수를 출력하고 둘째 줄 에는 회색 영역의 수를 출력한다.

입력 예	출력 예
8 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 1 0 0 0 1 1 1 1 0 1 0 0 1 1 1 1 0 0 1 1 1 1 1 1 0 0 1 1 1 1 1 1	9 7

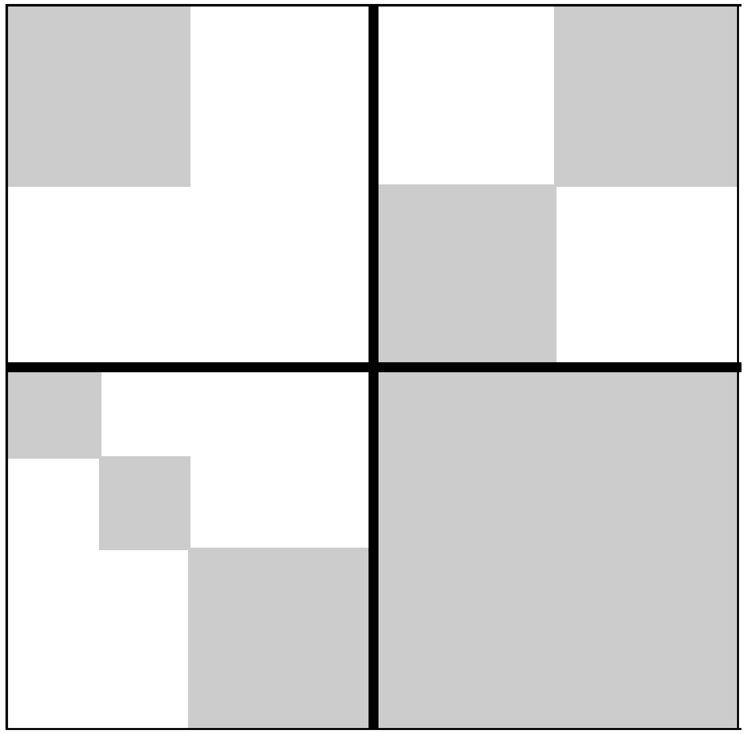
영역구분

아이디어



한 변의 길이가 8인 입력형태의 종이를 나타낸다. 모든 종이가 같은 모든 종이가 같은 색깔이 아니므로 4 등분으로 분할한다.

혼합색종이 = 1
하얀색종이 = 0
회색종이 = 0



한 번 분할된 한 번의 길이가 4인 종이인 상태.

좌표평면상에서의 사분면으로 볼 때, 4 사분면을 제외하면 모두 색깔이 한 가지가 아니므로 다시 길이가 2인 종이로 분할.

혼합색종이 = 3
하얀색종이 = 0
회색종이 = 1

영역구분

아이디어

이와 같은 단계로 처리하기 위해 다음과 같이 상태를 정의한다.

$f(a,\ b,\ n)$ = a행 b열을 왼쪽, 위 모서리로 하고 변의 길이가 n인 종이를 탐색

이 상태를 다음과 같은 관계식으로 정의할 수 있다.

$$f(a,\ b,\ n) = \begin{cases} 1 & (n=1,\ S[a][b]=1) \text{ 또는 } (S[a][b] \sim S[a+n-1][b+n-1] \text{ 까지 모두 } 1) \\ 0 & (n=1,\ S[a][b]=0) \text{ 또는 } (S[a][b] \sim S[a+n-1][b+n-1] \text{ 까지 모두 } 0) \\ f(a,\ b,\ \frac{n}{2}),\ f(a+\frac{n}{2},\ b,\ \frac{n}{2}), \\ f(a,\ b+\frac{n}{2},\ \frac{n}{2}),\ f(a+\frac{n}{2},\ b+\frac{n}{2}) & (\text{그 외}) \end{cases}$$

영역구분

https://onlinegdb.com/ryTDfvUiU

트로미노 문제

https://www.acmicpc.net/problem/14601

문제

오늘은 민규가 훈련소에 입소하는 날이다. 모든 행사를 마치고 생활관으로 돌아와서 쉬려는데 갑자기 교관이 들어오더니 민규의 이름을 부르는 것이 아닌가. 당황한 채로 따라갔더니 이번엔 김준서를 아느냐고 물어보았다. 그 녀석이 샤워실 바닥을 깔았는데, 배수구 위치까지 막아버렸다면서 같은 학교 출신인 민규가 다시 깔라는 것이었다.

어떻게 타일을 깔지 고민하던 민규는 샤워실의 구조가 정사각형이면서 한 변의 길이가 2의 제곱수라는 사실을 알아냈다. 준서는 여기까지만 고려해서 2x2 크기의 타일로 바닥을 전부 채운 것 같은데, 문제는 이렇게 하면 배수구가 있어야 할 위치를 비울 수가 없다는 것이다. 이런저런 방법을 생각하다가 4칸을 차지하는 정사각형 타일 대신 3칸을 차지하는 ㄱ자 모양의 타일을 사용하면 될 것 같다는 느낌을 받았다.

그런데 ㄱ자 타일을 어떻게 채워야 할까? 생각하다 지친 민규는 여러분에게 이 방법을 찾아달라고 부탁했다. 첫날부터 생활관에서 밤을 새우는 일이 없도록 여러분이 도와주자.

입력

첫 번째 줄에는 바닥의 한 변의 길이를 표현하는 자연수 K($1 \leq K \leq 7$) 가 주어진다. 이때 바닥의 크기는 2^K 가 됨에 유의하라. 두 번째 줄에는 배수구의 위치를 나타내는 자연수 x, y ($1 \leq x, y \leq 2^K$)가 공백으로 분리돼서 주어진다. **좌표는 왼쪽 위가 (0.0) 오른쪽 아래가 (2^k-1, 2^k-1)이다**

출력

각 타일마다 고유한 번호를 매긴 타일의 배치도를 출력한다. 각 타일의 번호에는 19000 이하의 자연수만을 사용해야 한다. 배수구가 있는 위치는 -1로 표시한다. 가능한 답 중 하나만 출력하면 된다.

만약 알맞게 타일을 배치하는 방법이 존재하지 않는다면 -1을 출력한다.

트로미노 문제

- 출력 예시

```
1 1 1


 1   1
 1  -1
```

```
2 1 1


 3   3   2   2
 3  -1   1   2
 4   1   1   5
 4   4   5   5
```

```
2 3 2
2 2 4 4
2 1 1 4
3 1 5 -1
3 3 5 5
```

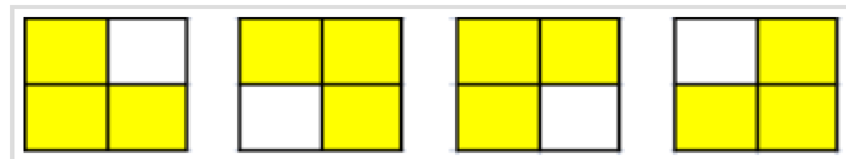
```
3 5 7
3 3 5 5 13 13 15 15
3 2 2 5 13 12 12 15
4 2 6 6 14 14 12 16
4 4 6 1 1 14 16 16
8 8 10 1 18 18 20 20
8 7 10 10 18 17 17 20
9 7 7 11 19 19 17 21
9 9 11 11 19 -1 21 21
```

트로미노 문제

아이디어

문제 : 2^n * 2^n 크기(n은 자연수)의 체스판이 있다. 그런데 이 체스판 중 1칸은 쓸 수 없다. 쓸 수 없는 칸의 위치가 주어질 때, 나머지 칸들을 모두 L 모양의 tromino로만 채우는 방법을 찾아야 한다.

L 모양의 tromino는 다음과 같이 생겼다.

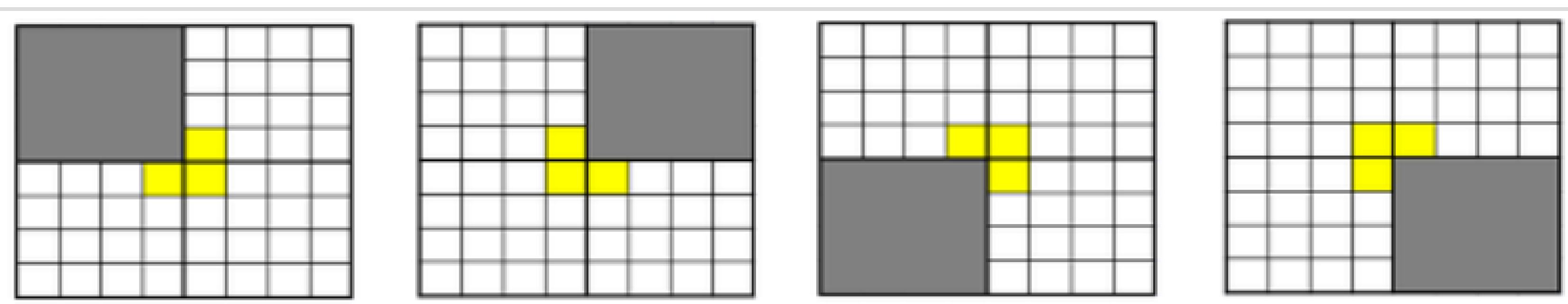


채우는 방법은 다음과 같다.

1. 먼저 2^n * 2^n 체스판을 사분면으로 나눈다. 편의상 사분면은 다음과 같이 번호를 붙인다.

2. 각각의 사분면은 2^(n-1) * 2^(n-1) 크기이다. 사분면 중에 쓸 수 없는 칸(Blank)이 있는 사분면을 제외한 나머지 3개의 사분면에 대해, 3개 사분면을 모두 걸치는 tromino를 놓는다.

아래 그림에서 회색 부분은 Blank를 포함한 사분면의 영역이다. 노란색 영역은 이번에 놓아야 할 타일의 위치이다.



1사분면에 Blank 존재 **2사분면에 Blank 존재** **3사분면에 Blank 존재** **4사분면에 Blank 존재**

3. 이제 나누어진 4개의 2^(n-1) * 2^(n-1) 사분면들에 대해 똑같이 과정 1~2번을 반복한다.

트로미노 문제

아이디어

$f(x, y, n)$ = (x, y)를 시작으로 하고 2^n의 크기를 갖는 바닥에 trimino를 놓는 함수

$f(x, y, n) =$

- -1 — $n < 1$ 일때
- $f(x, y, n-1)$, $f((x+2^n)/2+1, y, n-1)$, $f(x, (y+2^n)/2+1, n-1)$, $f((x+2^n)/2+1, (y+2^n)/2+1, n-1)$, — $n \geq 1$ 일때

  Blank가 존재하면 tromino를 놓는다.

  - 1사분면 $(x+1, y)$ $(x, y+1)$ $(x+1, y+1)$
  - 2사분면 (x, y) $(x, y+1)$ $(x+1, y+1)$
  - 3사분면 (x, y) $(x+1, y)$ $(x+1, y+1)$
  - 4사분면 (x, y) $(x+1, y)$ $(x, y+1)$

트로미노 문제

https://onlinegdb.com/SyoniyRsL

삼성 SW 역량 테스트 기출 문제 (쉬운 문제)

https://www.acmicpc.net/problem/13458

문제

총 N개의 시험장이 있고, 각각의 시험장마다 응시자들이 있다. i번 시험장에 있는 응시자의 수는 A_i명이다.

감독관은 총감독관과 부감독관으로 두 종류가 있다. 총감독관은 한 방에서 감시할 수 있는 응시자의 수가 B명이고, 부감독관은 한 방에서 감시할 수 있는 응시자의 수가 C명이다.

각각의 시험장에 총감독관은 오직 1명만 있어야 하고, 부감독관은 여러 명 있어도 된다.

각 시험장마다 응시생들을 모두 감시해야 한다. 이때, 필요한 감독관 수의 최솟값을 구하는 프로그램을 작성하시오.

입력

첫째 줄에 시험장의 개수 N($1 \le N \le 1{,}000{,}000$)이 주어진다.

둘째 줄에는 각 시험장에 있는 응시자의 수 A_i ($1 \le A_i \le 1{,}000{,}000$)가 주어진다.

셋째 줄에는 B와 C가 주어진다. ($1 \le B, C \le 1{,}000{,}000$)

출력

각 시험장마다 응시생을 모두 감독하기 위해 필요한 감독관의 최소 수를 출력한다.

예제 입력 1 복사

```
1
1
1 1
```

예제 출력 1 복사

```
1
```

삼성 SW 역량 테스트 기출 문제 (쉬운 문제)

예제 입력 2 복사

```
3
3 4 5
2 2
```

예제 출력 2 복사

```
7
```

예제 입력 3 복사

```
5
1000000 1000000 1000000 1000000 1000000
5 7
```

예제 출력 3 복사

```
714290
```

예제 입력 4 복사

```
5
10 9 10 9 10
7 20
```

예제 출력 4 복사

```
10
```

예제 입력 5 복사

```
5
10 9 10 9 10
7 2
```

예제 출력 5 복사

```
13
```

삼성 SW 역량 테스트 기출 문제 (쉬운 문제)

아이디어

a. 각각의 시험장에 총감독관이 1명 씩 있어야한다.

=> 모든 시험장에서 총감독관이 감독할 수 있는 응시자의 수를 뺀다.

b. 부감독관은 여러명 있어도 상관없다.

=> 총감독관이 감시 할 수 있는 인원을 뺀 상태의 시험장의 남은 사람이 있을때,

부감독관이 감독할 수 있는 경우의 수를 뺀다. (나눴을때 몫으로 구한다.)

딱 떨어지는 경우(나머지가 0인 경우)를 생각해준다.

삼성 SW 역량 테스트 기출 문제 (쉬운 문제)

https://onlinegdb.com/Bkj3GD8jU