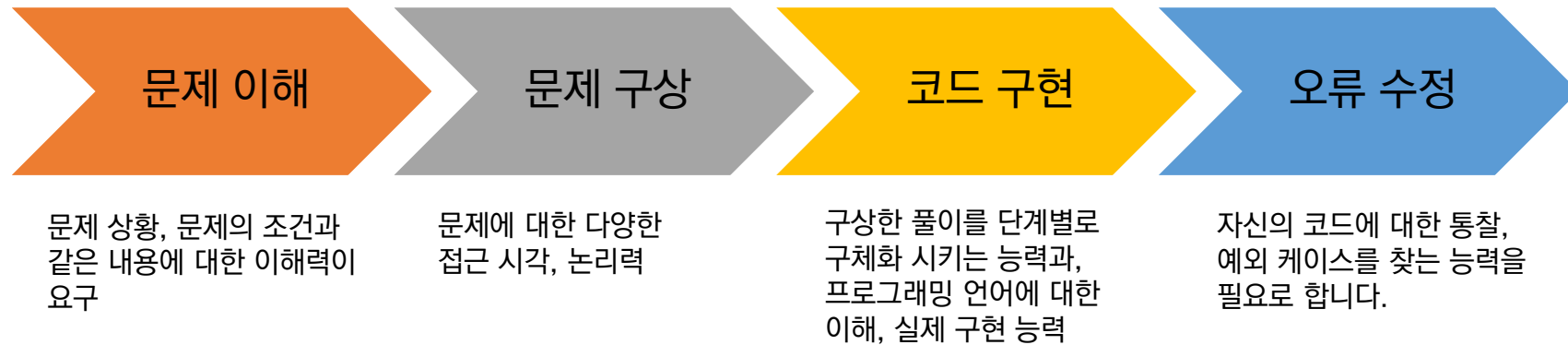


2021 여름학기 동국대학교 SW역량강화캠프

1일차. 구현 역량 강화

알고리즘 문제해결을 위한 과정

- 알고리즘들의 난이도는 그렇게 높지 않지만, 그 알고리즘들을 이용해 풀이를 구상하는 논리력이 중요
- 알고리즘 문제를 푸는 과정을 크게 4단계로 나누면 다음과 같이 나눌 수 있음

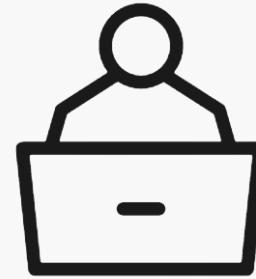


● 구현

구현이란 머리 속에 설계해놓은 추상적 알고리즘을 실제로 코드로 표현하는 것을 의미



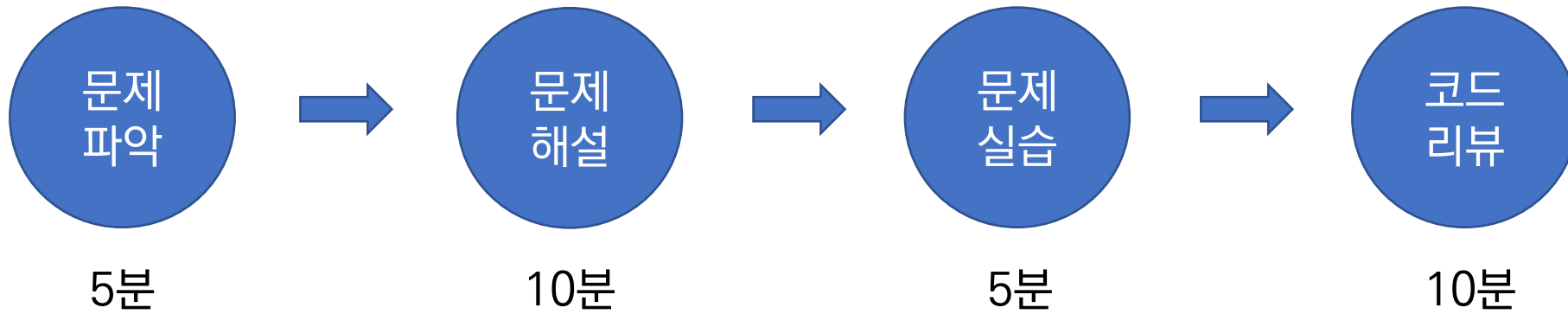
추상적 알고리즘



구현!

- 구현은 모든 프로그래밍의 기본이자, 코딩테스트에서 절반 이상을 차지하는 핵심!
- 알고리즘을 잘 알더라도 구현 역량이 없다면 이후 난이도 있는 알고리즘 문제들을 해결하는데도 한계에 부딪칠 수 밖에 없음
- ▶ 구현과 연결되어 있는 주요 개념: 조건문, 반복문, 배열 등 기초 문법

- 대표 문제 학습(수업당 4~5문제)



●울타리 색칠하기(1144)

| 문제

존과 윌리는 집을 둘러싼 울타리를 색칠하려고 한다.

울타리는 일직선 모양으로, 제일 왼쪽에는 0 제일 오른쪽에는 100이라는 수가 적혀 있다.

존은 (a, b)구간을 색칠하고 윌리는 (c, d)구간을 색칠했을때, 총 색칠한 구간의 길이를 구하자.

만약 존이 (7, 10)구간을 색칠하고 윌리가 (4, 8)구간을 색칠하였다면 총 색칠된 구간의 길이는 6이다.

| 입력

첫 줄에는 존이 색칠한 울타리의 범위 (a, b)가 주어진다.

두 번째 줄에는 윌리가 색칠한 울타리의 범위 (c, d)가 주어진다.

단, $a < b$, $c < d$ 이며 모든 수는 100 이하이다.

| 출력

페인트가 칠해진 울타리의 길이를 출력하여라.

● 코드를 짜기 전에..

문제를 정확하게 이해했는지 확인

예시 입력과 예시 출력을 활용

직접 계산해서, 정답과 일치하는 지 확인.

예제1 입력

7 10
4 8

예제1 출력

6

● 문제의 제한 조건 확인

주어진 입력의 범위

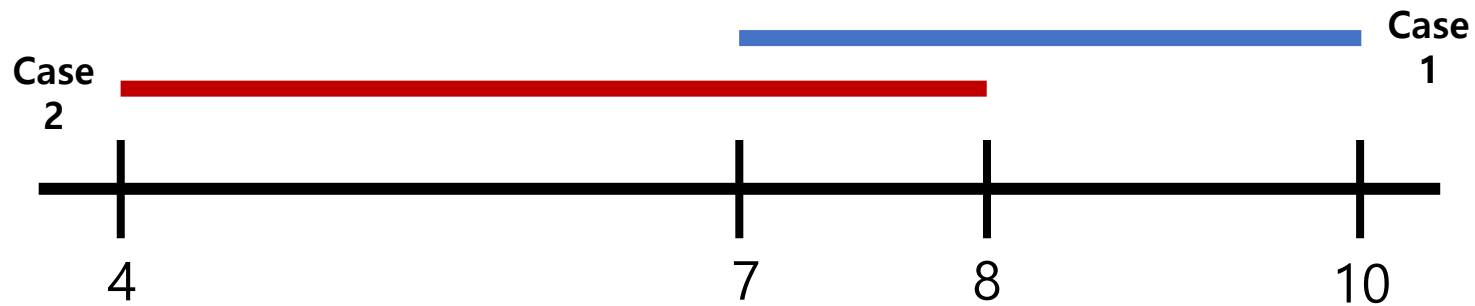
출력 조건

주어진 시간 제한(1초)

입력

첫 줄에는 존이 색칠한 울타리의 범위 (a, b)가 주어진다.
두 번째 줄에는 윌리가 색칠한 울타리의 범위 (c, d)가 주어진다.

단, $a < b$, $c < d$ 이며 모든 수는 100 이하이다.



예제1 입력

복사

7 10

4 8

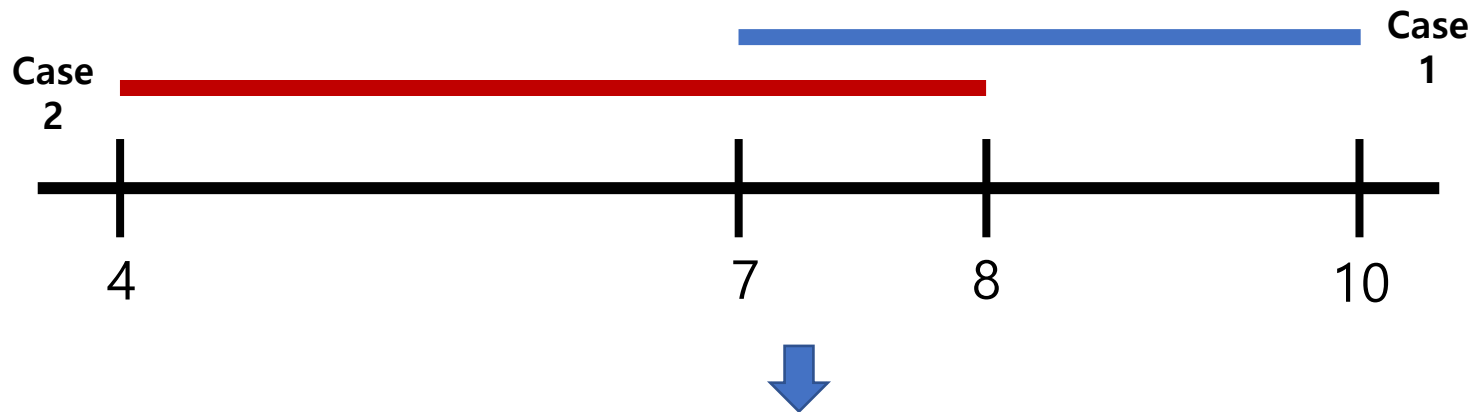
예제1 출력

복사

6

●배열을 이용한 문제 해결

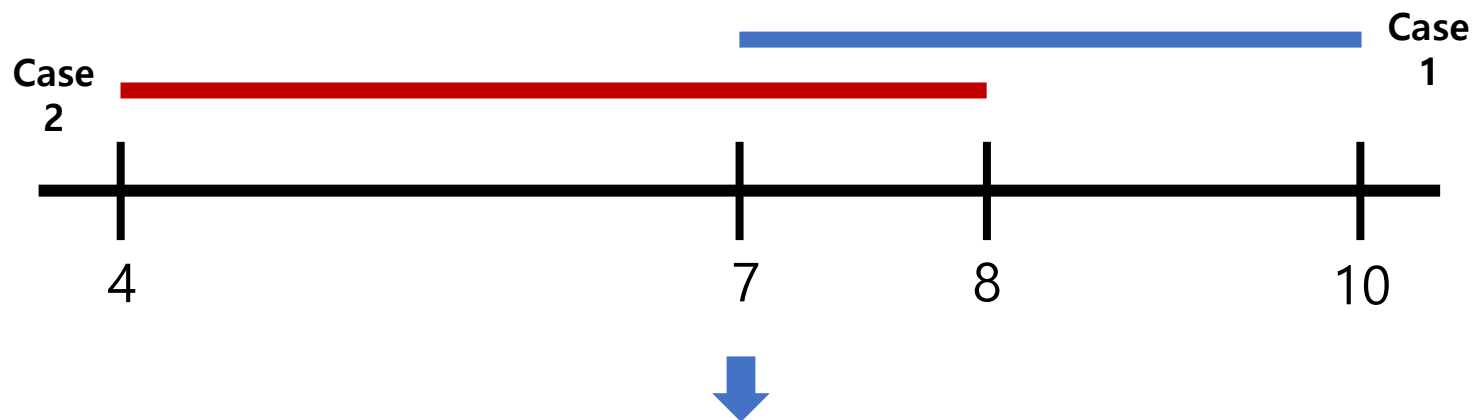
- ▶ 전체 울타리 구간 $[0, 100]$ 을 100개의 길이 1짜리 구간으로 나누어 배열에 대응
- ▶ $arr[0] : [0,1]$, $arr[1] : [1,2]$, , $arr[99] : [99,100]$
- ▶ 울타리 구간 $[a,b] = arr[a] \sim arr[b-1]$



arr[]	0	1	2	3	4	5	6	7	8	9	10
case1	0	0	0	0	0	0	0	0	0	0	0

● 배열을 이용한 문제 해결

- ▶ 울타리 구간을 색칠할 때, 대응하는 배열의 칸에 표시
- ▶ 총 색칠된 구간의 길이 = 표시된 배열의 칸 수



arr[]	0	1	2	3	4	5	6	7	8	9	10
case1	0	0	0	0	0	0	0	1	1	1	0
case2	0	0	0	0	1	1	1	1	0	0	0

```
for(int i = 0; i < 100; i++) x[i] = false;
for(int i = a; i < b; i++) {
    x[i] = true;
}
for(int i = c; i < d; i++) {
    x[i] = true;
}

int ans = 0;
for(int i = 0; i < 100; i++) {
    if (x[i]) ans++;
}
```

●라이프가드(1118)

문제

존은 윌리와 북극곰 친구들이 뛰어놀 수 있도록 수영장을 만들어주었다.

알다시피 남극이기 때문에 그냥 얼음을 적당히 깎 것만 수영장이기 때문에 상당히 위험하다.

안전성을 위해 존은 라이프가드를 몇 명 고용하여 윌리와 친구들의 안전을 지켜왔었지만, 최근 재정난으로 인해 한 명의 라이프가드를 해고할 예정이다.

사려깊은 존은 아무나 자르지 않고, 그나마 안전한 시간(한 명의 라이프가드라도 있는 시간) 가장 길도록 라이프가드를 해고할 예정이다.

```
3
5 9
1 4
3 7
```

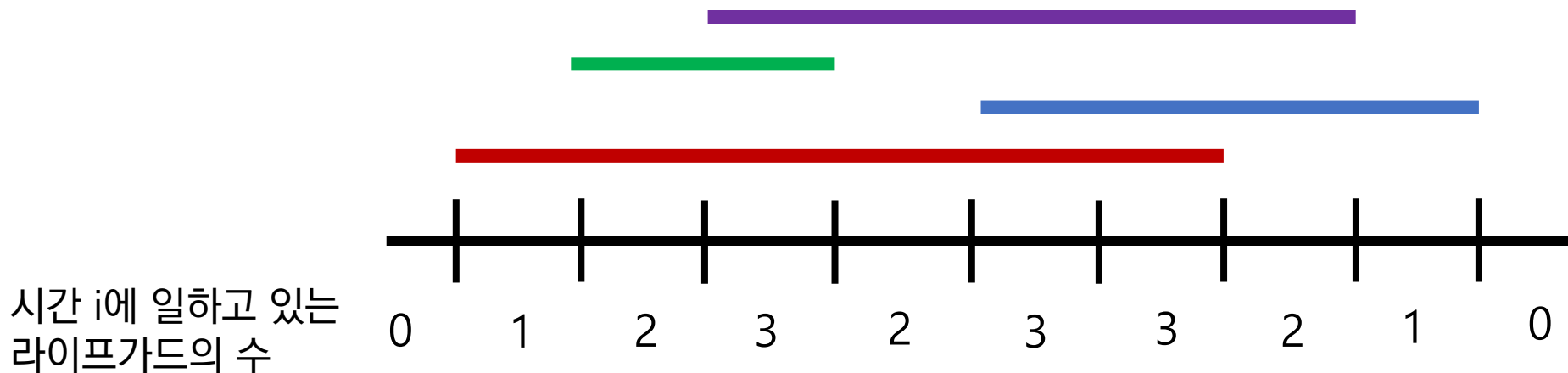
예를 들어 위 입력에서는 1번 라이프가드는 5분에서 9분까지(4분만 일하는 것이다!), 2번은 1분에서 4분까지, 3번은 3분에서 7분까지 근무를 하고 있고, 3번 라이프가드를 해고하게 되면 안전한 시간이 7분이 되어 최대이다.

존이 한 명의 라이프가드를 해고할 때, 가능한 가장 긴 안전한 시간을 출력하자.

● 배열을 이용한 문제 해결

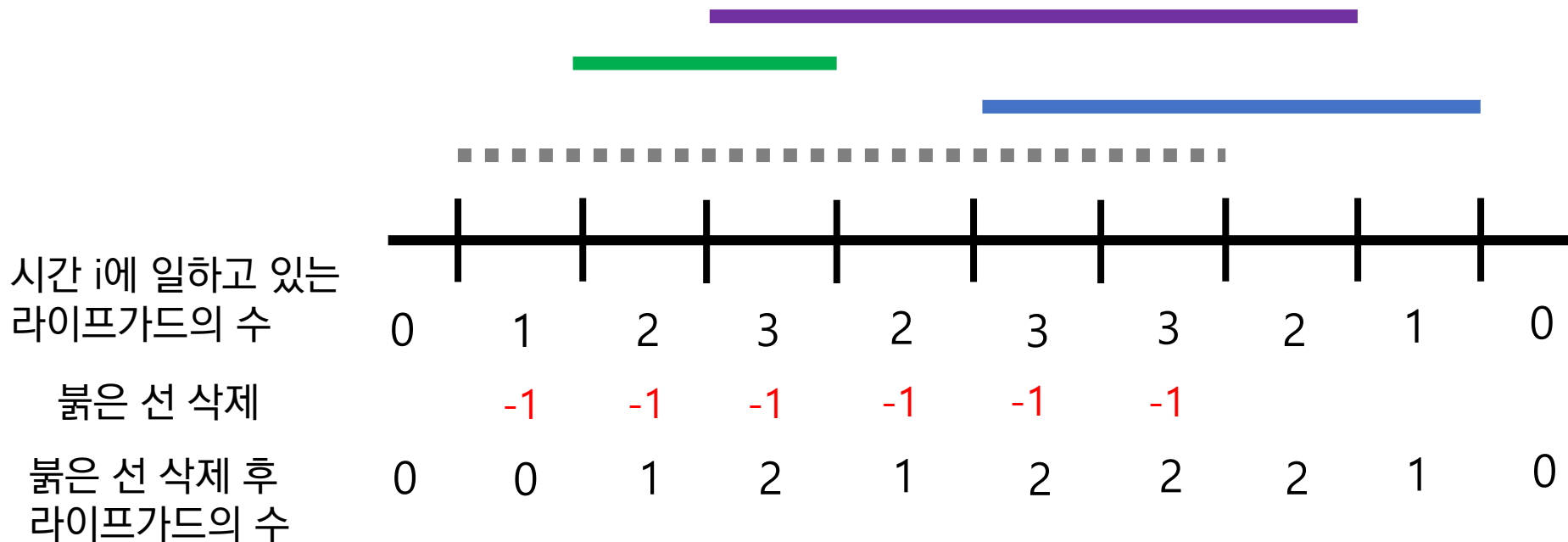
▶ 시간을 1분 단위로 쪼개어 배열에 대응시킨다.

▶ $arr[i] = [i, i+1]$ 분에 일하고 있는 라이프가드의 수



● 배열을 이용한 문제 해결

- ▶ 직원을 해고하면, 그 사람이 일하던 시간 동안에는 $arr[i]$ 가 1 줄어든다.
- ▶ 총 안전한 시간 = 라이프가드가 1명 이상 일하고 있는 시간 = $arr[i]$ 가 1 이상인 개수



```
int[] arr = new int[MAXN]; // arr[i] = i분대에 일하고 있는 라이프가드의 수
int answer = 0;

for (int i = 0; i < N; i++) {
    for (int j = brr[i][0]; j < brr[i][1]; j++) { // i번 라이프가드는 brr[i][0]분부터 brr[i][1]분 전까지 알했다.
        arr[j]++;
    }
}
```

```
for (int i = 0; i < N; i++) // i번 사람을 해고했을 때 안전한 시간을 계산
{
    // i번 직원을 해고하면 i번 직원의 근무시간에는 라이프가드의 수가 1 줄어든다.
    for (int j = brr[i][0]; j < brr[i][1]; j++)
        arr[j]--;

    int safetime = 0; // 안전한 시간을 count
    for (int j = 0; j < MAXN; j++) {
        if (arr[j] != 0) // 일하고 있는 라이프가드의 수가 0이 아니라면 이 시간대는 안전한 시간이다.
            safetime++;
    }

    // 안전한 시간의 최대값을 구한다.
    if (answer < safetime)
        answer = safetime;

    // arr 배열을 다시 원래대로
    for (int j = brr[i][0]; j < brr[i][1]; j++)
        arr[j]++;
}
```

●방 배정 (656)

문제

정보 초등학교에서는 단체로 2박 3일 수학여행을 가기로 했다. 여러 학년이 같은 장소로 수학여행을 가려고 하는데 1학년부터 6학년까지 학생들이 묵을 방을 배정해야 한다. 1~2학년은 남학생, 여학생 구별 없이 방을 배정할 수 있으며 3~6학년은 남학생은 남학생끼리, 여학생은 여학생끼리 방을 배정해야 한다. 또한 1~2학년은 학년 구별 없이 같은 방에 배정할 수 있으며 마찬가지로 3~4학년도, 5~6학년도 각각 학년 구별 없이 방을 배정할 수 있다. 물론 한 방에 한 명만 배정되는 것도 가능하다.

한 방에 배정할 수 있는 최대 인원 수 K 가 주어졌을 때, 조건에 맞게 모든 학생을 배정하기 위해 필요한 방의 최소 개수를 구하는 프로그램을 작성하시오.

예를 들어, 수학여행을 가는 학생이 다음과 같다면 $K = 2$ 일 때 9개의 방이 필요하다.

학년	여학생	남학생
1학년	영희	동호, 동진
2학년	혜진, 상희	경수
3학년	경희	동수, 상철, 철복
4학년		달호
5학년	정숙	호동, 건우
6학년	수지	동건

- 배열을 이용한 문제 해결

- ▶ 같은 방을 배정받을 수 있는 그룹의 사람 수를 센다.
- ▶ {1,2학년}, {3,4학년 남자}, {3,4학년 여자}, {5,6학년 남자}, {5,6학년 여자}
- ▶ 5개의 그룹의 인원수를 `arr[0] ~ arr[4]`에 저장

● 배열을 이용한 문제 해결

- ▶ X명을 한 방에 K명씩 넣으면, 총 몇 개의 방이 필요할까
- ▶ K명씩 (X/K) 개의 방에 배정, 남는 학생이 있다면 추가로 1개의 방이 더 필요
- ▶ $X\%K$ 가 0 이라면 (X/K) 개, 0이 아니라면 $(X/K)+1$ 개의 방이 필요하다.

```
while (N-- > 0) {  
    st = new StringTokenizer(br.readLine());  
    int S = Integer.parseInt(st.nextToken());  
    int Y = Integer.parseInt(st.nextToken());  
  
    if ((Y == 1) || (Y == 2)) arr[0]++; // 1학년이거나 2학년이면 arr[0]  
    else if ((Y == 3) || (Y == 4)) {  
        if (S == 1) arr[1]++; // 3,4학년 남자는 arr[1]  
        else arr[2]++; // 3,4학년 여자는 arr[2]  
    }  
    else {  
        if (S == 1) arr[3]++; // 5,6학년 남자는 arr[3]  
        else arr[4]++; // 5,6학년 여자는 arr[4]  
    }  
}
```

```
int ans = 0;
for (int i = 0; i < 5; i++) {
    if(arr[i] % K == 0) ans += arr[i] / K;
    else ans += arr[i] / K + 1;

    // ans += (arr[i] + K - 1) / K; // 위의 if 문을 이렇게 대체할 수 있습니다.
}
```

● 이미지 확대 (350)

| 문제

원시인 선진이는 컴퓨터에 저장된 이미지들을 보기가 너무 어렵다고 한다

이를 안 선진이의 착한 친구 현재는 선진이를 위해 모든 이미지들을 x 배 확대 시켜주는 프로그램을 만들어주려고 한다.

기존의 이미지와 선진이가 확대를 원하는 배율 x 가 주어질 때, 기존의 이미지를 x 배 만큼 확대해 주는 프로그램을 작성하여라.

| 입력

첫째 줄에 기존 이미지의 크기를 나타내는 두 정수 n, m 이 공백을 사이에 두고 주어진다. ($1 \leq n, m \leq 100$)

다음 n 개의 줄에 걸쳐 $n*m$ 크기의 이미지가 주어진다. (이는 숫자, 알파벳, 기호 등 공백을 제외한 모든 문자들이 주어질 수 있다.)

마지막 줄에는 원하는 배율 x 가 주어진다. ($1 \leq x \leq 10$)

| 출력

입력으로 주어진 $n*m$ 크기의 이미지를 가로, 세로 x 배씩 확대한 $nx*mx$ 크기의 이미지를 출력하여라.

- 반복문을 이용한 문제 해결
 - ▶ 가로로 K배 확대하기
 - ▶ $ABC \rightarrow AAABBBCCC$
 - ▶ 문자를 출력할 때 K번 반복해서 출력

- 반복문을 이용한 문제 해결

- ▶ 세로로 K배 확대하기

- ▶ $ABC \rightarrow \begin{matrix} ABC \\ ABC \\ ABC \end{matrix}$

- ▶ 한 줄을 출력할 때 K번 반복해서 출력

```
for(int i = 1; i <= N ; i++){ // i번째 줄을 출력할 때
    for(int I = 1; I <= k; I++){ // k번 반복한다. (세로로 k배)
        for(int j = 1; j <= M; j++){// i번째 줄을 출력할 때 문자열의 j번째 문자를
            for(int J = 1; J <= k; J++){ // k번 반복하여 출력한다. (가로로 k배)
                sb.append(arr[i][j]);
            }
        }
        sb.append("\n");
    }
}
```