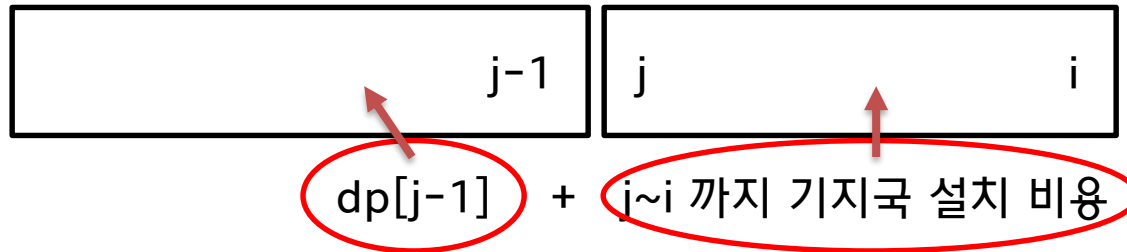


# 기지국

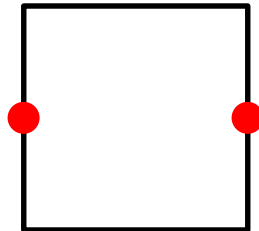
<https://www.acmicpc.net/problem/2300>

- 기지국은 항상 정사각형 모양이다.
- 기지국은 그룹을 지어 연속으로 지어야 최소가 된다.
- $dp[i]$  =  $i$ 번째 건물까지 통신 범위에 포함시키는 최소 설치 비용



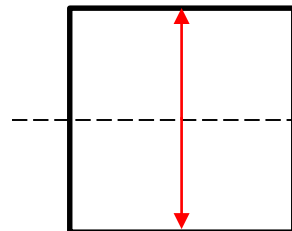
- $dp[i] = dp[j-1] + (j \sim i \text{ 까지 기지국 설치 비용})$
- 기지국 설치 비용

① 가로 폭이 최대

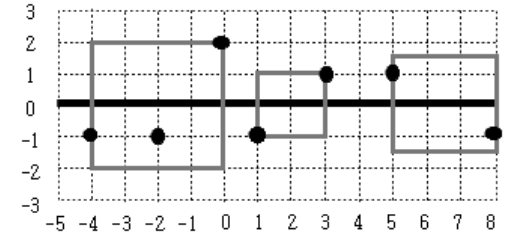


$$A[i].x - A[j].x$$

② 세로 폭이 최대



$$\max(A[j].y) (j = 1 \sim i)$$



# 기지국

<https://www.acmicpc.net/problem/2300>

```
int solve(int i) {  
    if (i == 0) return 0;  
  
    if (dp[i] != -1) return dp[i];  
  
    dp[i] = 1e9;  
    int width = 0, height = 0;  
    for (int j = i; j >= 1; --j) {  
        width = A[i].x - A[j].x;  
        height = max(height, A[j].y);  
        dp[i] = min(dp[i], solve(j - 1) + max(width, 2 * height));  
    }  
    return dp[i];  
}
```

- 역순으로 탐색을 해야 올바른 답을 얻을 수 있다. 그 이유는 구간  $[j, i]$  에서 가장 y좌표가 큰 기지국을 알 수 있어야 하는데 순방향으로 탐색하면  $[j, i]$ 가 아닌  $[1, j]$  구간의 y좌표가 큰 기지국을 알 수 있게 되기 때문이다.  
만약 순방향으로 탐색을 하려면 Binary Index Tree를 이용하여 구간 최대 값을 구하면 된다.