

백준 1388 바닥 장식

202020988 조아영

001 문제 소개

002 아이디어

003 코드 설명

문제 소개

‘-’와 ‘|’로 이루어진 바닥 장식 모양이 주어진다.

두 개의 ‘-’가 인접해 있고, 같은 행에 있다면, 두 개는 같은 나무 판자이고,
두 개의 ‘|’가 인접해 있고, 같은 열에 있다면, 두 개는 같은 나무 판자이다.

바닥을 장식하는데 필요한 나무 판자의 개수를 출력하는 프로그램을 작성

입력 : 방 바닥의 세로 크기 N , 가로 크기 M (50 이하의 정수)
 N 개의 줄에 M 개의 바닥 장식 모양 입력 : ‘-’ 또는 ‘|’

출력 : 바닥을 장식하는데 필요한 나무 판자의 개수

아이디어

반복문을 사용해서 인접한 바닥 장식 모양을 비교하면서
일일이 나무 판자의 개수를 셈

DFS / BFS

코드 설명 1

```
import sys
```

세로, 가로 크기 입력

```
N, M = map(int, sys.stdin.readline().split())
```

```
floor = []
```

```
for _ in range(N):
```

바닥 장식의 모양 입력

```
    floor.append(list(sys.stdin.readline()))
```

```
temp = []
```

```
for _ in range(M):
```

```
    temp.append('\n')
```

```
floor.append(temp)
```

바닥 장식의 최종 모양 저장

```
-----\n-----\n-----\n-----\n-----\n\n\n\n\n\n
```

코드 설명 1

```
cnt = 0
for i in range(N):
    for j in range(M+1):
        if floor[i][j] == '-':
            if floor[i][j] != floor[i][j+1]:
                cnt += 1

for i in range(N+1):
    for j in range(M):
        if floor[i][j] == '|':
            if floor[i][j] != floor[i+1][j]:
                cnt += 1

print(cnt)
```

바닥 장식 모양 '-'인
나무 판자의 개수 셈

바닥 장식 모양 '|'인
나무 판자의 개수 셈

나무 판자의 개수 출력

코드 설명 2

```
if __name__ == "__main__":  
    N, M = map(int, sys.stdin.readline().split()) 세로, 가로의 크기 입력  
  
    floor = []  
    for _ in range(N):  
        floor.append(list(sys.stdin.readline())) 바닥 장식의 모양 입력  
  
    visited = [[False for _ in range(M)] for _ in range(N)]  
    cnt = 0 방문을 체크할 2차원 배열, 나무 판자의 개수를 셀 변수 초기화  
  
    for i in range(N):  
        for j in range(M):  
            if visited[i][j] == False:  
                dfs(i,j) 방문하지 않은 곳의 경우 dfs 함수 호출  
                cnt += 1 나무 판자의 개수 셈  
  
    print(cnt) 나무 판자의 개수 출력
```

코드 설명 2

```
def dfs(x,y):  
    visited[x][y] = True 방문 표시  
    d = [1, -1] 인접한 곳 확인 하기 위함  
  
    if floor[x][y] == '-':  
        for i in range(2): 양 옆의 인접한 바닥 장식 확인  
            ny = y + d[i]  
            같은 모양의 바닥 장식이고, 방문하지 않은 곳인 경우  
            if 0<= ny < M and floor[x][ny] == '-' and visited[x][ny] == False:  
                dfs(x, ny) dfs 함수 호출  
  
    if floor[x][y] == '|':  
        for i in range(2): 위 아래의 인접한 바닥 장식 확인  
            nx = x + d[i]  
  
            if 0<= nx < N and floor[nx][y] == '|' and visited[nx][y] == False:  
                dfs(nx, y)
```


감사합니다