

多源BFS: [1020. 飞地的数量](#)

```

1  class Solution {
2      int dx[4] = {1,-1,0,0};
3      int dy[4] = {0,0,1,-1};
4  public:
5      int numEnclaves(vector<vector<int>>& grid) {
6          int m = grid.size(), n = grid[0].size();
7          vector<vector<bool>> vis(m, vector<bool>(n));
8          queue<pair<int, int>> q;
9
10         for (int i = 0; i < m; i++) {
11             for (int j = 0; j < n; j++) {
12                 if (i == 0 || i == m - 1 || j == 0 || j == n - 1) {
13                     if (grid[i][j] == 1) {
14                         q.push({i, j});
15                         vis[i][j] = true;
16                     }
17                 }
18             }
19         }
20         while(q.size())
21         {
22             auto [a,b] = q.front();
23             q.pop();
24             for(int i = 0; i < 4; i++)
25             {
26                 int x = a + dx[i];
27                 int y = b + dy[i];
28                 if(x >= 0 && x < m && y >= 0 && y < n && grid[x][y] == 1 && !vis[x][y])
29                 {
30                     vis[x][y] = true;
31                     q.push({x,y});
32                 }
33             }
34         }
35         int ret = 0;
36         for(int i = 0; i < m; i++)
37         {
38             for(int j = 0; j < n; j++)
39             {

```

```
40         if(grid[i][j] == 1 && !vis[i][j])
41             ret++;
42     }
43 }
44 return ret;
45 }
46 };
```