

그레프 1 - 도전

스 코드

최백준 choi@startlink.io



C++14

```
1 #include <cstdio>
2 #include <queue>
3 using namespace std;
4 int a[100][100];
5 int g[100][100];
6 int d[100][100];
7 int dx[] = {0,0,1,-1};
8 int dy[] = {1,-1,0,0};
9 int main() {
10     int n;
11     scanf("%d",&n);
12     for (int i=0; i<n; i++) {
13         for (int j=0; j<n; j++) {
14             scanf("%d",&a[i][j]);
15         }
16     }
17     int cnt = 0;
18     for (int i=0; i<n; i++) {
19         for (int j=0; j<n; j++) {
20             if (a[i][j] == 1 && g[i][j] == 0) {
21                 g[i][j] = ++cnt;
22                 queue<pair<int,int>> q;
23                 q.push(make_pair(i,j));
24                 while (!q.empty()) {
25                     int x = q.front().first;
26                     int y = q.front().second;
27                     q.pop();
28                     for (int k=0; k<4; k++) {
29                         int nx = x+dx[k];
30                         int ny = y+dy[k];
31                         if (0 <= nx && nx < n && 0 <= ny && ny < n) {
32                             if (a[nx][ny] == 1 && g[nx][ny] == 0) {
33                                 g[nx][ny] = cnt;
34                                 q.push(make_pair(nx,ny));
35                             }
36                         }
37                     }
38                 }
39             }
40         }
41     }
42     int ans = -1;
43     for (int l=1; l<=cnt; l++) {
44         queue<pair<int,int>> q;
45         for (int i=0; i<n; i++) {
46             for (int j=0; j<n; j++) {
47                 d[i][j] = -1;
48                 if (g[i][j] == l) {
49                     q.push(make_pair(i,j));
50                     d[i][j] = 0;
51                 }
52             }
53         }
54         while (!q.empty()) {
55             int x = q.front().first;
56             int y = q.front().second;
57             q.pop();
58             for (int k=0; k<4; k++) {
59                 int nx = x+dx[k];
60                 int ny = y+dy[k];
61                 if (0 <= nx && nx < n && 0 <= ny && ny < n) {
62                     if (d[nx][ny] == -1) {
63                         d[nx][ny] = d[x][y] + 1;
64                         q.push(make_pair(nx,ny));
65                     }
66                 }
67             }
68         }
69         for (int i=0; i<n; i++) {
70             for (int j=0; j<n; j++) {
71                 if (a[i][j] == 1 && g[i][j] != l) {
72                     if (ans == -1 || d[i][j]-1 < ans) {
73                         ans = d[i][j]-1;
74                     }
75                 }
76             }
77         }
78     }
79     printf("%d\n",ans);
80     return 0;
81 }
```

결과

메모리

시간

코드 길이

맞았습니다!!

1344 KB

844 ms

2380 B

C++14

```
1 #include <cstdio>
2 #include <queue>
3 using namespace std;
4 int a[100][100];
5 int g[100][100];
6 int d[100][100];
7 int dx[] = {0,0,1,-1};
8 int dy[] = {1,-1,0,0};
9 int main() {
10     int n;
11     scanf("%d",&n);
12     for (int i=0; i<n; i++) {
13         for (int j=0; j<n; j++) {
14             scanf("%d",&a[i][j]);
15         }
16     }
17     int cnt = 0;
18     for (int i=0; i<n; i++) {
19         for (int j=0; j<n; j++) {
20             if (a[i][j] == 1 && g[i][j] == 0) {
21                 g[i][j] = ++cnt;
22                 queue<pair<int,int>> q;
23                 q.push(make_pair(i,j));
24                 while (!q.empty()) {
25                     int x = q.front().first;
26                     int y = q.front().second;
27                     q.pop();
28                     for (int k=0; k<4; k++) {
29                         int nx = x+dx[k];
30                         int ny = y+dy[k];
31                         if (0 <= nx && nx < n && 0 <= ny && ny < n) {
32                             if (a[nx][ny] == 1 && g[nx][ny] == 0) {
33                                 g[nx][ny] = cnt;
34                                 q.push(make_pair(nx,ny));
35                             }
36                         }
37                     }
38                 }
39             }
40         }
41     }
42     queue<pair<int,int>> q;
43     for (int i=0; i<n; i++) {
44         for (int j=0; j<n; j++) {
45             d[i][j] = -1;
46             if (a[i][j] == 1) {
47                 q.push(make_pair(i,j));
48                 d[i][j] = 0;
49             }
50         }
51     }
52     while (!q.empty()) {
53         int x = q.front().first;
54         int y = q.front().second;
55         q.pop();
56         for (int k=0; k<4; k++) {
57             int nx = x+dx[k];
58             int ny = y+dy[k];
59             if (0 <= nx && nx < n && 0 <= ny && ny < n) {
60                 if (d[nx][ny] == -1) {
61                     d[nx][ny] = d[x][y] + 1;
62                     g[nx][ny] = g[x][y];
63                     q.push(make_pair(nx,ny));
64                 }
65             }
66         }
67     }
68     int ans = -1;
69     for (int i=0; i<n; i++) {
70         for (int j=0; j<n; j++) {
71             for (int k=0; k<4; k++) {
72                 int x = i+dx[k];
73                 int y = j+dy[k];
74                 if (0 <= x && x < n && 0 <= y && y < n) {
75                     if (g[i][j] != g[x][y]) {
76                         if (ans == -1 || ans > d[i][j] + d[x][y]) {
77                             ans = d[i][j] + d[x][y];
78                         }
79                     }
80                 }
81             }
82         }
83     }
84     printf("%d\n",ans);
85     return 0;
86 }
```

BFS

(i,j) - (x,y)
d[i][j] + d[x][y]

0-4ms

결과	메모리	시간	코드 길이
맞았습니다!!	1344 KB	0 ms	2486 B

끝

코드 플러스

<https://code.plus>

- 슬라이드에 포함된 소스 코드를 보려면 "정보 수정 > 백준 온라인 저지 연동"을 통해 연동한 다음, "백준 온라인 저지"에 로그인해야 합니다.
- 강의 내용에 대한 질문은 코드 플러스의 "질문 게시판"에서 할 수 있습니다.
- 문제와 소스 코드는 슬라이드에 첨부된 링크를 통해서 볼 수 있으며, "백준 온라인 저지"에서 서비스됩니다.
- 슬라이드와 동영상 강의는 코드 플러스 사이트를 통해서만 볼 수 있으며, 동영상 강의의 녹화와 다운로드, 배포와 유통은 저작권법에 의해서 금지되어 있습니다.
- 다른 경로로 이 슬라이드나 동영상 강의를 본 경우에는 codeplus@startlink.io 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.