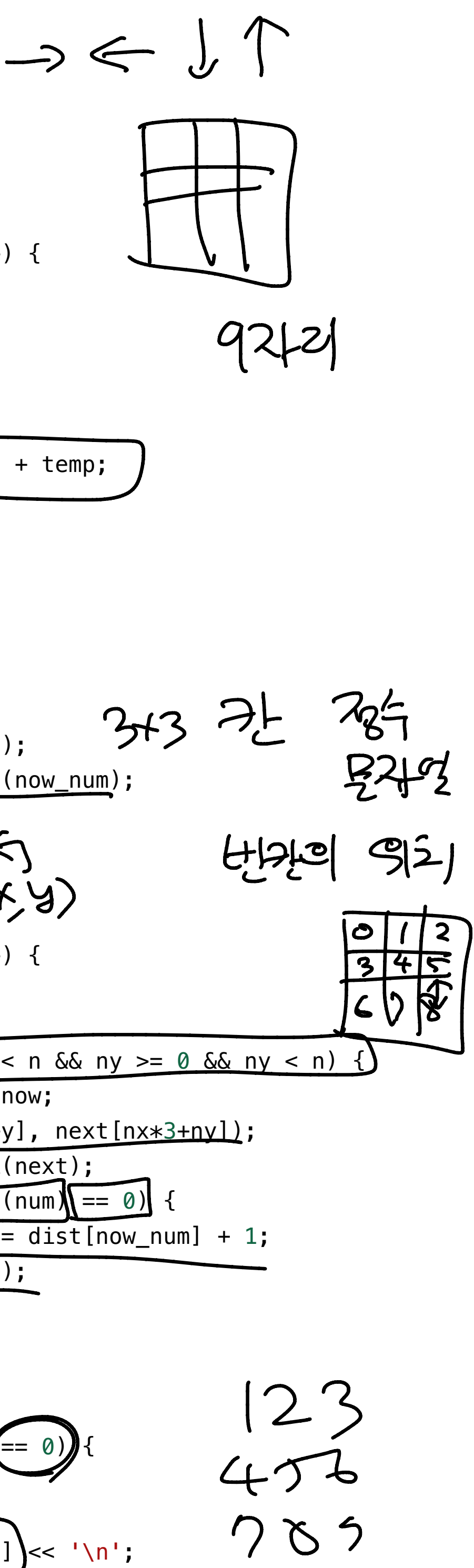


최백준 [choi@startlink.io](mailto:choi@startlink.io)



C++14

```
1 #include <iostream>
2 #include <queue>
3 #include <string>
4 #include <map>
5 using namespace std;
6 int dx[] = {0, 0, 1, -1};
7 int dy[] = {1, -1, 0, 0};
8 int main() {
9     int n = 3;
10    int start = 0;
11    for (int i=0; i<n; i++) {
12        for (int j=0; j<n; j++) {
13            int temp;
14            cin >> temp;
15            if (temp == 0) {
16                temp = 9;
17            }
18            start = start * 10 + temp;
19        }
20    }
21    queue<int> q;
22    map<int, int> dist;
23    dist[start] = 0;
24    q.push(start);
25    while (!q.empty()) {
26        int now_num = q.front();
27        string now = to_string(now_num);
28        q.pop();
29        int z = now.find('9');
30        int x = z/3;
31        int y = z%3;
32        for (int k=0; k<4; k++) {
33            int nx = x+dx[k];
34            int ny = y+dy[k];
35            if (nx >= 0 && nx < n && ny >= 0 && ny < n) {
36                string next = now;
37                swap(next[x*3+y], next[nx*3+ny]);
38                int num = stoi(next);
39                if (dist.count(num) == 0) {
40                    dist[num] = dist[now_num] + 1;
41                    q.push(num);
42                }
43            }
44        }
45    }
46    if (dist.count(123456789) == 0) {
47        cout << "-1" << '\n';
48    } else {
49        cout << dist[123456789] << '\n';
50    }
51    return 0;
52 }
```



결과	메모리	시간	코드 길이
맞았습니다!!	10440 KB	200 ms	1287 B

Java

```
1 import java.util.*;
2
3 public class Main {
4     public static final int[] dx = {0, 0, 1, -1};
5     public static final int[] dy = {1, -1, 0, 0};
6     public static void main(String args[]) {
7         Scanner sc = new Scanner(System.in);
8         int n = 3;
9         int start = 0;
10        for (int i=0; i<n; i++) {
11            for (int j=0; j<n; j++) {
12                int temp = sc.nextInt();
13                if (temp == 0) {
14                    temp = 9;
15                }
16                start = start * 10 + temp;
17            }
18        }
19        Queue<Integer> q = new LinkedList<Integer>();
20        HashMap<Integer, Integer> d = new HashMap<Integer, Integer>();
21        d.put(start, 0);
22        q.add(start);
23        while (!q.isEmpty()) {
24            int now_num = q.remove();
25            String now = Integer.toString(now_num);
26            int z = now.indexOf('9');
27            int x = z/3;
28            int y = z%3;
29            for (int k=0; k<4; k++) {
30                int nx = x+dx[k];
31                int ny = y+dy[k];
32                if (nx >= 0 && nx < n && ny >= 0 && ny < n) {
33                    StringBuilder next = new StringBuilder(now);
34                    char temp = next.charAt(x*3+y);
35                    next.setCharAt(x*3+y, next.charAt(nx*3+ny));
36                    next.setCharAt(nx*3+ny, temp);
37                    int num = Integer.parseInt(next.toString());
38                    if (!d.containsKey(num)) {
39                        d.put(num, d.get(now_num)+1);
40                        q.add(num);
41                    }
42                }
43            }
44        }
45        if (d.containsKey(123456789)) {
46            System.out.println(d.get(123456789));
47        } else {
48            System.out.println("-1");
49        }
50    }
51 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	77964 KB	808 ms	1742 B



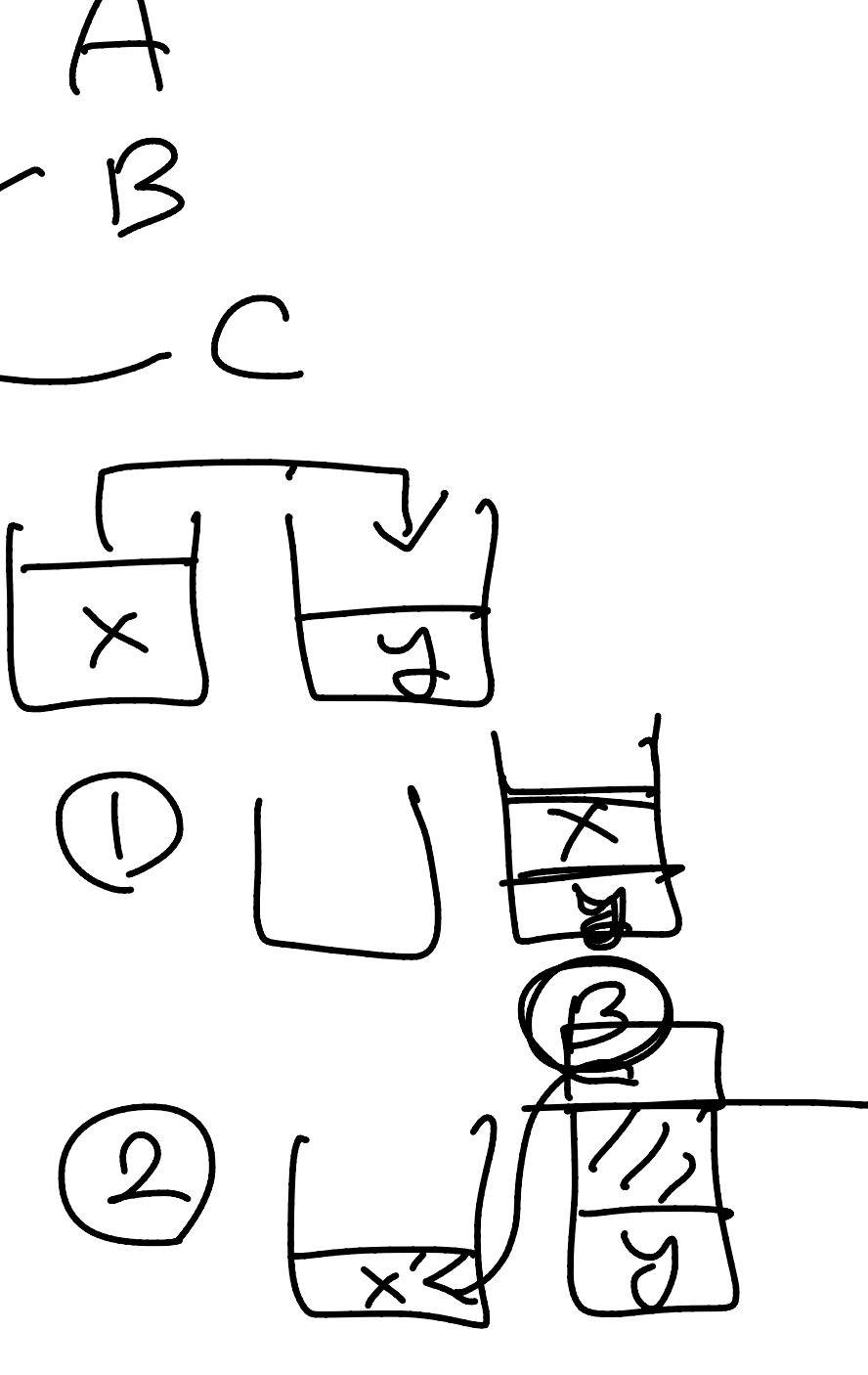
## C++14

```
1 #include <iostream>
2 #include <queue>
3 using namespace std;
4 bool ans[201];
5 bool check[201][201];
6 int cap[3];
7 int from[3] = {0, 0, 1, 1, 2, 2};
8 int to[3] = {1, 2, 0, 2, 0, 1};
9 int main() {
10     for (int i=0; i<3; i++) {
11         cin >> cap[i];
12     }
13     int sum = cap[2];
14     queue<pair<int,int>> q;
15     q.push(make_pair(0, 0));
16     check[0][0] = true;
17     ans[cap[2]] = true;
18     while (!q.empty()) {
19         int cur[3];
20         cur[0] = q.front().first;
21         cur[1] = q.front().second;
22         cur[2] = sum - cur[0] - cur[1];
23         q.pop();
24         for (int k=0; k<6; k++) {
25             int next[3] = {cur[0], cur[1], cur[2]};
26             next[to[k]] += next[from[k]];
27             next[from[k]] = 0;
28             if (next[to[k]] >= cap[to[k]]) {
29                 next[from[k]] = next[to[k]] - cap[to[k]];
30                 next[to[k]] = cap[to[k]];
31             }
32             if (!check[next[0]][next[1]]) {
33                 check[next[0]][next[1]] = true;
34                 q.push(make_pair(next[0], next[1]));
35                 if (next[0] == 0) {
36                     ans[next[2]] = true;
37                 }
38             }
39         }
40     }
41     for (int i=0; i<=cap[2]; i++) {
42         if (ans[i]) {
43             cout << i << ' ';
44         }
45     }
46     cout << '\n';
47     return 0;
48 }
```

A B C  
0 1 2

## C++11

```
1 #include <iostream>
2 #include <queue>
3 using namespace std;
4 bool ans[201];
5 bool check[201][201];
6 int main() {
7     int a, b, c;
8     cin >> a >> b >> c;
9     int sum = c;
10    queue<pair<int,int>> q;
11    q.push(make_pair(0, 0));
12    check[0][0] = true;
13    ans[c] = true;
14    while (!q.empty()) {
15        int x = q.front().first;
16        int y = q.front().second;
17        int z = sum - x - y;
18        q.pop();
19        int nx, ny, nz;
20        nx = x;
21        ny = y;
22        nz = z;
23        // x -> y
24        ny += nx;
25        nx = 0;
26        if (ny >= b) {
27            nx = ny-b;
28            ny = b;
29        }
30        if (!check[nx][ny]) {
31            check[nx][ny] = true;
32            q.push(make_pair(nx,ny));
33            if (nx == 0) {
34                ans[nz] = true;
35            }
36        }
37
38        nx = x;
39        ny = y;
40        nz = z;
41        // x -> z
42        nz += nx;
43        nx = 0;
44        if (nz >= c) {
45            nx = nz-c;
46            nz = c;
47        }
48        if (!check[nx][ny]) {
49            check[nx][ny] = true;
50            q.push(make_pair(nx,ny));
51            if (nx == 0) {
52                ans[nz] = true;
53            }
54        }
55
56        nx = x;
57        ny = y;
58        nz = z;
59        // y -> x
60        nx += ny;
61        ny = 0;
62        if (nx >= a) {
63            ny = nx-a;
64            nx = a;
65        }
66        if (!check[nx][ny]) {
67            check[nx][ny] = true;
68            q.push(make_pair(nx,ny));
69            if (nx == 0) {
70                ans[nz] = true;
71            }
72        }
73
74        nx = x;
75        ny = y;
76        nz = z;
77        // y -> z
78        nz += ny;
79        ny = 0;
80        if (nz >= c) {
81            ny = nz-c;
82            nz = c;
83        }
84        if (!check[nx][ny]) {
85            check[nx][ny] = true;
86            q.push(make_pair(nx,ny));
87            if (nx == 0) {
88                ans[nz] = true;
89            }
90        }
91
92        nx = x;
93        ny = y;
94        nz = z;
95        // z -> x
96        nx += nz;
97        nz = 0;
98        if (nx >= a) {
99            nz = nx-a;
100            nx = a;
101        }
102        if (!check[nx][ny]) {
103            check[nx][ny] = true;
104            q.push(make_pair(nx,ny));
105            if (nx == 0) {
106                ans[nz] = true;
107            }
108        }
109
110        nx = x;
111        ny = y;
112        nz = z;
113        // z -> y
114        ny += nz;
115        nz = 0;
116        if (ny >= b) {
117            nz = ny-b;
118            ny = b;
119        }
120        if (!check[nx][ny]) {
121            check[nx][ny] = true;
122            q.push(make_pair(nx,ny));
123            if (nx == 0) {
124                ans[nz] = true;
125            }
126        }
127    }
128    for (int i=0; i<=c; i++) {
129        if (ans[i]) {
130            cout << i << ' ';
131        }
132    }
133    cout << '\n';
134    return 0;
135 }
```



## Java

```
1 import java.util.*;
2
3 class Pair implements Comparable<Pair> {
4     final int first;
5     final int second;
6
7     Pair(int first, int second) {
8         this.first = first;
9         this.second = second;
10    }
11
12    @Override
13    public int compareTo(Pair pair) {
14        if (this.first < pair.first) {
15            return -1;
16        }
17        if (this.first > pair.first) {
18            return 1;
19        }
20        if (this.second < pair.second) {
21            return -1;
22        }
23        if (this.second > pair.second) {
24            return 1;
25        }
26        return 0;
27    }
28
29    public boolean equals(Object object) {
30        if (object instanceof Pair) {
31            Pair pair = (Pair)object;
32            return this.first == pair.first && this.second == pair.second;
33        }
34        return false;
35    }
36
37    public int hashCode() {
38        int n = 3;
39        n = 19 * n + this.first;
40        n = 19 * n + this.second;
41        return n;
42    }
43 }
44
45 public class Main {
46     public static final int[] from = {0, 0, 1, 1, 2, 2};
47     public static final int[] to = {1, 2, 0, 2, 0, 1};
48     public static void main(String args[]) {
49         Scanner sc = new Scanner(System.in);
50         int[] cap = new int[3];
51         for (int i=0; i<3; i++) {
52             cap[i] = sc.nextInt();
53         }
54         int sum = cap[2];
55         boolean[][] check = new boolean[201][201];
56         boolean[] ans = new boolean[201];
57         Queue<Pair> q = new LinkedList<Pair>();
58         q.add(new Pair(0, 0));
59         check[0][0] = true;
60         ans[cap[2]] = true;
61         while (!q.isEmpty()) {
62             int[] cur = new int[3];
63             Pair p = q.peek();
64             cur[0] = p.first;
65             cur[1] = p.second;
66             cur[2] = sum - cur[0] - cur[1];
67             q.remove();
68             for (int k=0; k<6; k++) {
69                 int[] next = {cur[0], cur[1], cur[2]};
70                 next[to[k]] += next[from[k]];
71                 next[from[k]] = 0;
72                 if (next[to[k]] >= cap[to[k]]) {
73                     next[from[k]] = next[to[k]] - cap[to[k]];
74                     next[to[k]] = cap[to[k]];
75                 }
76                 if (!check[next[0]][next[1]]) {
77                     check[next[0]][next[1]] = true;
78                     q.add(new Pair(next[0], next[1]));
79                     if (next[0] == 0) {
80                         ans[next[2]] = true;
81                     }
82                 }
83             }
84         }
85         for (int i=0; i<=cap[2]; i++) {
86             if (ans[i]) {
87                 System.out.print(i + " ");
88             }
89         }
90         System.out.println();
91     }
92 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	2028 KB	0 ms	2736 B



## C++14

```
1 #include <iostream>
2 #include <vector>
3 #include <string>
4 #include <algorithm>
5 #include <deque>
6 #include <tuple>
7 using namespace std;
8 int dx[] = {0, 0, 1, -1};
9 int dy[] = {1, -1, 0, 0};
10 vector<vector<int>> bfs(vector<string>&a, int x, int y) {
11     int n = a.size();
12     int m = a[0].size();
13     vector<vector<int>> d(n, vector<int>(m));
14     for (int i=0; i<n; i++) {
15         for (int j=0; j<m; j++) {
16             d[i][j] = -1;
17         }
18     }
19     deque<pair<int,int>> q;
20     q.push_back(make_pair(x, y));
21     d[x][y] = 0;
22     while (!q.empty()) {
23         tie(x,y) = q.front(); q.pop_front();
24         for (int k=0; k<4; k++) {
25             int nx = x+dx[k];
26             int ny = y+dy[k];
27             if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
28             if (d[nx][ny] != -1) continue;
29             if (a[nx][ny] == '*') continue;
30             if (a[nx][ny] == '#') {
31                 d[nx][ny] = d[x][y] + 1;
32                 q.push_back(make_pair(nx,ny));
33             } else {
34                 d[nx][ny] = d[x][y];
35                 q.push_front(make_pair(nx,ny));
36             }
37         }
38     }
39     return d;
40 }
41 int main() {
42     int t;
43     cin >> t;
44     while (t--) {
45         int n, m;
46         cin >> n >> m;
47         vector<string> a(n+2);
48         for (int i=1; i<=n; i++) {
49             cin >> a[i];
50             a[i] = "." + a[i] + ".";
51         }
52         n += 2;
53         m += 2;
54         for (int j=0; j<m; j++) {
55             a[0] += ".";
56             a[n-1] += ".";
57         }
58         vector<vector<int>> d0 = bfs(a, 0, 0);
59         int x1, y1, x2, y2;
60         x1 = y1 = x2 = y2 = -1;
61         for (int i=0; i<n; i++) {
62             for (int j=0; j<m; j++) {
63                 if (a[i][j] == '$') {
64                     if (x1 == -1) {
65                         x1 = i;
66                         y1 = j;
67                     } else {
68                         x2 = i;
69                         y2 = j;
70                     }
71                 }
72             }
73         }
74         vector<vector<int>> d1 = bfs(a, x1, y1);
75         vector<vector<int>> d2 = bfs(a, x2, y2);
76         int ans = n*m;
77         for (int i=0; i<n; i++) {
78             for (int j=0; j<m; j++) {
79                 if (a[i][j] == '*') continue;
80                 int cur = d0[i][j] + d1[i][j] + d2[i][j];
81                 if (a[i][j] == '#') cur -= 2;
82                 if (ans > cur) ans = cur;
83             }
84         }
85         cout << ans << '\n';
86     }
87     return 0;
88 }
```

Handwritten notes and diagrams:

- Arrows pointing to `dx` and `dy` arrays with the text "213 (x,y)".
- A circle around `d[x][y] = 0;` with the text "알 (5)".
- A circle around `d0 = bfs(a, 0, 0);` with the text "(0,0) → [diagram]".
- A diagram showing three circles labeled 1, 2, and 3. Circle 1 has a yellow arrow pointing to a blue square. Circle 2 has a cyan arrow pointing to the same blue square. Circle 3 has a red arrow pointing to the blue square. The blue square has a red arrow pointing to it from the right, labeled "64 n".
- A circle around the calculation `int cur = d0[i][j] + d1[i][j] + d2[i][j];` with the text "D/2".

결과	메모리	시간	코드 길이
맞았습니다!!	2136 KB	16 ms	2410 B

## Java

```
1 import java.util.*;
2 class Pair {
3     int x, y;
4     Pair(int x, int y) {
5         this.x = x;
6         this.y = y;
7     }
8 }
9 public class Main {
10     public static int[] dx = {1, -1, 0, 0};
11     public static int[] dy = {0, 0, 1, -1};
12     public static int[][] bfs(String[] a, int x, int y) {
13         int n = a.length;
14         int m = a[0].length();
15         int[][] d = new int[n][m];
16         for (int i=0; i<n; i++) {
17             for (int j=0; j<m; j++) {
18                 d[i][j] = -1;
19             }
20         }
21         ArrayDeque<Pair> deque = new ArrayDeque<Pair>();
22         deque.add(new Pair(x, y));
23         d[x][y] = 0;
24         while (!deque.isEmpty()) {
25             Pair p = deque.poll();
26             x = p.x;
27             y = p.y;
28             for (int k=0; k<4; k++) {
29                 int nx = x+dx[k];
30                 int ny = y+dy[k];
31                 if (nx < 0 || nx >= n || ny < 0 || ny >= m) continue;
32                 if (d[nx][ny] != -1) continue;
33                 char c = a[nx].charAt(ny);
34                 if (c == '*') continue;
35                 if (c == '#') {
36                     d[nx][ny] = d[x][y] + 1;
37                     deque.addLast(new Pair(nx, ny));
38                 } else {
39                     d[nx][ny] = d[x][y];
40                     deque.addFirst(new Pair(nx, ny));
41                 }
42             }
43         }
44         return d;
45     }
46     public static void main(String[] args) {
47         Scanner sc = new Scanner(System.in);
48         int t = sc.nextInt();
49         while (t-- > 0) {
50             int n = sc.nextInt();
51             int m = sc.nextInt();
52             sc.nextLine();
53             String[] a = new String[n+2];
54             for (int i=1; i<=n; i++) {
55                 a[i] = sc.nextLine();
56                 a[i] = "." + a[i] + ".";
57             }
58             n += 2;
59             m += 2;
60             a[0] = a[n-1] = "";
61             for (int j=0; j<m; j++) {
62                 a[0] += ".";
63                 a[n-1] += ".";
64             }
65             int[][] d0 = bfs(a, 0, 0);
66             int x1, y1, x2, y2;
67             x1 = y1 = x2 = y2 = -1;
68             for (int i=0; i<n; i++) {
69                 for (int j=0; j<m; j++) {
70                     if (a[i].charAt(j) == '$') {
71                         if (x1 == -1) {
72                             x1 = i;
73                             y1 = j;
74                         } else {
75                             x2 = i;
76                             y2 = j;
77                         }
78                     }
79                 }
80             }
81             int[][] d1 = bfs(a, x1, y1);
82             int[][] d2 = bfs(a, x2, y2);
83             int ans = n*m;
84             for (int i=0; i<n; i++) {
85                 for (int j=0; j<m; j++) {
86                     char c = a[i].charAt(j);
87                     if (c == '*') continue;
88                     int cur = d0[i][j] + d1[i][j] + d2[i][j];
89                     if (c == '#') cur -= 2;
90                     if (ans > cur) ans = cur;
91                 }
92             }
93             System.out.println(ans);
94         }
95     }
96 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	31660 KB	324 ms	2999 B



## C++14

```
1 #include <cstdio>
2 #include <cstring>
3 #include <queue>
4 using namespace std;
5 char a[111][111];
6 bool c[111][111];
7 bool key[111];
8 int dx[] = {0, 0, 1, -1};
9 int dy[] = {1, -1, 0, 0};
10 int main() {
11     int t;
12     scanf("%d",&t);
13     while (t-->0) {
14         int n, m;
15         memset(a,0,sizeof(a));
16         scanf("%d %d",&n,&m);
17         for (int i=2; i<n+2; i++) {
18             scanf("%s",a[i]+2);
19         }
20         n += 4;
21         m += 4;
22         for (int i=0; i<n; i++) {
23             a[i][0] = '*';
24             a[i][1] = '.';
25             a[i][m-2] = '.';
26             a[i][m-1] = '*';
27         }
28         for (int j=1; j<m-1; j++) {
29             a[0][j] = '*';
30             a[1][j] = '.';
31             a[n-2][j] = '.';
32             a[n-1][j] = '*';
33         }
34         memset(key,false,sizeof(key));
35         char temp[111];
36         scanf("%s",temp);
37         int len = strlen(temp);
38         if (temp[0] != '0') {
39             for (int i=0; i<len; i++) {
40                 key[temp[i]-'a'] = true;
41             }
42         }
43         int ans = 0;
44         memset(c,false,sizeof(c));
45         queue<pair<int,int>> q;
46         queue<pair<int,int>> door[26];
47         q.push(make_pair(1,1));
48         c[1][1] = true;
49         while (!q.empty()) {
50             int x = q.front().first;
51             int y = q.front().second;
52             q.pop();
53             for (int k=0; k<4; k++) {
54                 int nx = x+dx[k];
55                 int ny = y+dy[k];
56                 if (c[nx][ny]) {
57                     continue;
58                 }
59                 char w = a[nx][ny];
60                 if (w == '*') {
61                     continue;
62                 }
63                 c[nx][ny] = true;
64                 if (w == '.') {
65                     q.push(make_pair(nx,ny));
66                 } else if (w == '$') {
67                     ans += 1;
68                     q.push(make_pair(nx,ny));
69                 } else if (w >= 'A' && w <= 'Z') {
70                     if (key[w-'A']) {
71                         q.push(make_pair(nx,ny));
72                     } else {
73                         door[w-'A'].push(make_pair(nx,ny));
74                     }
75                 } else if (w >= 'a' && w <= 'z') {
76                     q.push(make_pair(nx,ny));
77                     if (!key[w-'a']) {
78                         key[w-'a'] = true;
79                         while (!door[w-'a'].empty()) {
80                             q.push(door[w-'a'].front());
81                             door[w-'a'].pop();
82                         }
83                     }
84                 }
85             }
86         }
87         printf("%d\n",ans);
88     }
89     return 0;
90 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1252 KB	4 ms	2564 B

## Java

```
1 import java.util.*;
2 public class Main {
3     static char[][] a = new char[111][111];
4     static boolean[][] c = new boolean[111][111];
5     static boolean[] key = new boolean[111];
6     static int[] dx = {0,0,1,-1};
7     static int[] dy = {1,-1,0,0};
8     public static void main(String args[]) {
9         Scanner sc = new Scanner(System.in);
10        int t = sc.nextInt();
11        while (t-->0) {
12            int n = sc.nextInt();
13            int m = sc.nextInt();
14            for (int i=0; i<n+2; i++) {
15                for (int j=0; j<n+2; j++) {
16                    a[i][j] = 0;
17                }
18            }
19            for (int i=2; i<n+2; i++) {
20                String line = sc.next();
21                for (int j=0; j<m; j++) {
22                    a[i][j+2] = line.charAt(j);
23                }
24            }
25            n += 4;
26            m += 4;
27            for (int i=0; i<n; i++) {
28                a[i][0] = '*';
29                a[i][1] = '.';
30                a[i][m-2] = '.';
31                a[i][m-1] = '*';
32            }
33            for (int j=1; j<m-1; j++) {
34                a[0][j] = '*';
35                a[1][j] = '.';
36                a[n-2][j] = '.';
37                a[n-1][j] = '*';
38            }
39            Arrays.fill(key, false);
40            String temp = sc.next();
41            int len = temp.length();
42            if (temp.charAt(0) != '0') {
43                for (int i=0; i<len; i++) {
44                    key[temp.charAt(i)-'a'] = true;
45                }
46            }
47            int ans = 0;
48            for (int i=0; i<n+2; i++) {
49                Arrays.fill(c[i], false);
50            }
51            Queue<Integer> q = new LinkedList<>();
52            Queue<Integer>[] door = new LinkedList[26];
53            for (int i=0; i<26; i++) {
54                door[i] = new LinkedList<Integer>();
55            }
56            q.add(1); q.add(1);
57            c[1][1] = true;
58            while (!q.isEmpty()) {
59                int x = q.remove();
60                int y = q.remove();
61                for (int k=0; k<4; k++) {
62                    int nx = x+dx[k];
63                    int ny = y+dy[k];
64                    if (c[nx][ny]) {
65                        continue;
66                    }
67                    char w = a[nx][ny];
68                    if (w == '*') {
69                        continue;
70                    }
71                    c[nx][ny] = true;
72                    if (w == '.') {
73                        q.add(nx); q.add(ny);
74                    } else if (w == '$') {
75                        ans += 1;
76                        q.add(nx); q.add(ny);
77                    } else if (w >= 'A' && w <= 'Z') {
78                        if (key[w-'A']) {
79                            q.add(nx); q.add(ny);
80                        } else {
81                            door[w-'A'].add(nx);
82                            door[w-'A'].add(ny);
83                        }
84                    } else if (w >= 'a' && w <= 'z') {
85                        q.add(nx); q.add(ny);
86                        if (!key[w-'a']) {
87                            key[w-'a'] = true;
88                            while (!door[w-'a'].isEmpty()) {
89                                q.add(door[w-'a'].remove());
90                            }
91                        }
92                    }
93                }
94            }
95            System.out.println(ans);
96        }
97    }
98 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	23244 KB	284 ms	3363 B



C++14

```
1 #include <iostream>
2 #include <algorithm>
3 #include <tuple>
4 #include <queue>
5 #include <string>
6 #include <vector>
7 using namespace std;
8 int dx[] = {0,0,1,-1};
9 int dy[] = {1,-1,0,0};
10 vector<vector<int>> bfs(vector<string> &a, int sx, int sy) {
11     int n = a.size();
12     int m = a[0].size();
13     vector<vector<int>> dist(n, vector<int>(m,-1));
14     queue<pair<int,int>> q;
15     q.push(make_pair(sx,sy));
16     dist[sx][sy] = 0;
17     while (!q.empty()) {
18         int x, y;
19         tie(x,y) = q.front(); q.pop();
20         for (int k=0; k<4; k++) {
21             int nx = x+dx[k];
22             int ny = y+dy[k];
23             if (0 <= nx && nx < n && 0 <= ny && ny < m) {
24                 if (dist[nx][ny] == -1 && a[nx][ny] != 'x') {
25                     dist[nx][ny] = dist[x][y] + 1;
26                     q.push(make_pair(nx,ny));
27                 }
28             }
29         }
30     }
31     return dist;
32 }
33 int main() {
34     while (true) {
35         int n, m;
36         cin >> m >> n;
37         if (n == 0 && m == 0) break;
38         vector<string> a(n);
39         for (int i=0; i<n; i++) {
40             cin >> a[i];
41         }
42         vector<pair<int,int>> b(1);
43         for (int i=0; i<n; i++) {
44             for (int j=0; j<m; j++) {
45                 if (a[i][j] == 'o') {
46                     b[0] = make_pair(i,j);
47                 } else if (a[i][j] == '*') {
48                     b.push_back(make_pair(i,j));
49                 }
50             }
51         }
52         int l = b.size();
53         vector<vector<int>> d(l, vector<int>(l));
54         bool ok = true;
55         for (int i=0; i<l; i++) {
56             auto dist = bfs(a,b[i].first,b[i].second);
57             for (int j=0; j<l; j++) {
58                 d[i][j] = dist[b[j].first][b[j].second];
59                 if (d[i][j] == -1) {
60                     ok = false;
61                 }
62             }
63         }
64         if (ok == false) {
65             cout << -1 << '\n';
66             continue;
67         }
68         vector<int> p(l-1);
69         for (int i=0; i<l-1; i++) {
70             p[i] = i+1;
71         }
72         int ans = -1;
73         do {
74             int now = d[0][p[0]];
75             for (int i=0; i<l-2; i++) {
76                 now += d[p[i]][p[i+1]];
77             }
78             if (ans == -1 || ans > now) {
79                 ans = now;
80             }
81         } while(next_permutation(p.begin(), p.end()));
82         cout << ans << '\n';
83     }
84     return 0;
85 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1996 KB	52 ms	2346 B

Java

```
1 import java.util.*;
2 class Pair {
3     int first;
4     int second;
5     Pair(int first, int second) {
6         this.first = first;
7         this.second = second;
8     }
9 }
10 public class Main {
11     static final int[] dx = {0,0,1,-1};
12     static final int[] dy = {1,-1,0,0};
13     static int[][] bfs(String[] a, int sx, int sy) {
14         int n = a.length;
15         int m = a[0].length();
16         int[][] dist = new int[n][m];
17         for (int i=0; i<n; i++) {
18             for (int j=0; j<m; j++) {
19                 dist[i][j] = -1;
20             }
21         }
22         Queue<Pair> q = new LinkedList<Pair>();
23         q.add(new Pair(sx,sy));
24         dist[sx][sy] = 0;
25         while (!q.isEmpty()) {
26             Pair p = q.remove();
27             int x = p.first;
28             int y = p.second;
29             for (int k=0; k<4; k++) {
30                 int nx = x+dx[k];
31                 int ny = y+dy[k];
32                 if (0 <= nx && nx < n && 0 <= ny && ny < m) {
33                     if (dist[nx][ny] == -1 && a[nx].charAt(ny) != 'x') {
34                         dist[nx][ny] = dist[x][y] + 1;
35                         q.add(new Pair(nx,ny));
36                     }
37                 }
38             }
39         }
40         return dist;
41     }
42     static boolean next_permutation(int[] a) {
43         int i = a.length-1;
44         while (i > 0 && a[i-1] >= a[i]) {
45             i -= 1;
46         }
47         if (i <= 0) {
48             return false;
49         }
50         int j = a.length-1;
51         while (a[j] <= a[i-1]) {
52             j -= 1;
53         }
54         int temp = a[i-1];
55         a[i-1] = a[j];
56         a[j] = temp;
57         j = a.length-1;
58         while (i < j) {
59             temp = a[i];
60             a[i] = a[j];
61             a[j] = temp;
62             i += 1;
63             j -= 1;
64         }
65         return true;
66     }
67     public static void main(String args[]) {
68         Scanner sc = new Scanner(System.in);
69         while (true) {
70             int m = sc.nextInt();
71             int n = sc.nextInt();
72             if (n == 0 && m == 0) break;
73             String[] a = new String[n];
74             for (int i=0; i<n; i++) {
75                 a[i] = sc.next();
76             }
77             ArrayList<Pair> b = new ArrayList<>();
78             b.add(new Pair(0,0));
79             for (int i=0; i<n; i++) {
80                 for (int j=0; j<m; j++) {
81                     char x = a[i].charAt(j);
82                     if (x == 'o') {
83                         b.set(0, new Pair(i,j));
84                     } else if (x == '*') {
85                         b.add(new Pair(i,j));
86                     }
87                 }
88             }
89             int l = b.size();
90             int[][] d = new int[l][l];
91             boolean ok = true;
92             for (int i=0; i<l; i++) {
93                 int[][] dist = bfs(a, b.get(i).first, b.get(i).second);
94                 for (int j=0; j<l; j++) {
95                     d[i][j] = dist[b.get(j).first][b.get(j).second];
96                     if (d[i][j] == -1) {
97                         ok = false;
98                     }
99                 }
100             }
101             if (ok == false) {
102                 System.out.println(-1);
103                 continue;
104             }
105             int[] p = new int[l-1];
106             for (int i=0; i<l-1; i++) {
107                 p[i] = i+1;
108             }
109             int ans = -1;
110             do {
111                 int now = d[0][p[0]];
112                 for (int i=0; i<l-2; i++) {
113                     now += d[p[i]][p[i+1]];
114                 }
115                 if (ans == -1 || ans > now) {
116                     ans = now;
117                 }
118             } while(next_permutation(p));
119             System.out.println(ans);
120         }
121     }
122 }
123 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	27116 KB	640 ms	3614 B

C++14

```
1 #include <iostream>
2 #include <queue>
3 #include <tuple>
4 #include <string>
5 #include <algorithm>
6 using namespace std;
7 int dx[] = {0,0,1,-1};
8 int dy[] = {1,-1,0,0};
9 int main() {
10     int m, n;
11     cin >> m >> n;
12     vector<string> a(n);
13     int sx,sy,ex,ey;
14     sx=sy=ex=ey=-1;
15     for (int i=0; i<n; i++) {
16         cin >> a[i];
17         for (int j=0; j<m; j++) {
18             if (a[i][j] == 'C') {
19                 if (sx == -1) {
20                     sx = i;
21                     sy = j;
22                 } else {
23                     ex = i;
24                     ey = j;
25                 }
26             }
27         }
28     }
29     vector<vector<int>> d(n, vector<int>(m, -1));
30     queue<pair<int,int>> q;
31     d[sx][sy] = 0;
32     q.push(make_pair(sx,sy));
33     while (!q.empty()) {
34         int x, y;
35         tie(x,y) = q.front(); q.pop();
36         for (int k=0; k<4; k++) {
37             int nx = x+dx[k];
38             int ny = y+dy[k];
39             while (0 <= nx && nx < n && 0 <= ny && ny < m) {
40                 if (a[nx][ny] == '*') break;
41                 if (d[nx][ny] == -1) {
42                     d[nx][ny] = d[x][y] + 1;
43                     q.push(make_pair(nx,ny));
44                 }
45                 nx += dx[k];
46                 ny += dy[k];
47             }
48         }
49     }
50     cout << d[ex][ey]-1 << '\n';
51     return 0;
52 }
```

결과

맞았습니다!!

메모리

1992 KB

시간

0 ms

코드 길이

1301 B

Java

```
1 import java.util.*;
2 class Pair {
3     int first;
4     int second;
5     Pair(int first, int second) {
6         this.first = first;
7         this.second = second;
8     }
9 }
10 public class Main {
11     static final int[] dx = {0,0,1,-1};
12     static final int[] dy = {1,-1,0,0};
13     public static void main(String args[]) {
14         Scanner sc = new Scanner(System.in);
15         int m = sc.nextInt();
16         int n = sc.nextInt();
17         String[] a = new String[n];
18         int sx,sy,ex,ey;
19         sx=sy=ex=ey=-1;
20         for (int i=0; i<n; i++) {
21             a[i] = sc.next();
22             for (int j=0; j<m; j++) {
23                 if (a[i].charAt(j) == 'C') {
24                     if (sx == -1) {
25                         sx = i;
26                         sy = j;
27                     } else {
28                         ex = i;
29                         ey = j;
30                     }
31                 }
32             }
33         }
34         int[][] d = new int[n][m];
35         for (int i=0; i<n; i++) {
36             for (int j=0; j<m; j++) {
37                 d[i][j] = -1;
38             }
39         }
40         Queue<Pair> q = new LinkedList<Pair>();
41         q.add(new Pair(sx,sy));
42         d[sx][sy] = 0;
43         while (!q.isEmpty()) {
44             Pair p = q.remove();
45             int x = p.first;
46             int y = p.second;
47             for (int k=0; k<4; k++) {
48                 int nx = x+dx[k];
49                 int ny = y+dy[k];
50                 while (0 <= nx && nx < n && 0 <= ny && ny < m) {
51                     if (a[nx].charAt(ny) == '*') break;
52                     if (d[nx][ny] == -1) {
53                         d[nx][ny] = d[x][y] + 1;
54                         q.add(new Pair(nx,ny));
55                     }
56                     nx += dx[k];
57                     ny += dy[k];
58                 }
59             }
60         }
61         System.out.println(d[ex][ey]-1);
62     }
63 }
64 }
```

결과

맞았습니다!!

메모리

12716 KB

시간

184 ms

코드 길이

1836 B



C++14

```
1 #include <iostream>
2 #include <algorithm>
3 #include <vector>
4 #include <string>
5 #include <queue>
6 using namespace std;
7 int main() {
8     int t;
9     cin >> t;
10    while (t--) {
11        int n;
12        cin >> n;
13        vector<int> from(n,-1);
14        vector<int> how(n,-1);
15        vector<int> dist(n,-1);
16        queue<int> q;
17        q.push(1%n);
18        dist[1%n] = 0;
19        how[1%n] = 1;
20        while (!q.empty()) {
21            int now = q.front(); q.pop();
22            for (int i=0; i<=1; i++) {
23                int next = (now*10+i)%n;
24                if (dist[next] == -1) {
25                    dist[next] = (dist[now] + 1);
26                    from[next] = now;
27                    how[next] = i;
28                    q.push(next);
29                }
30            }
31        }
32        if (dist[0] == -1) {
33            cout << "BRAK" << '\n';
34        } else {
35            string ans = "";
36            for (int i=0; i!=-1; i=from[i]) {
37                ans += to_string(how[i]);
38            }
39            reverse(ans.begin(),ans.end());
40            cout << ans << '\n';
41        }
42    }
43    return 0;
44 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	2180 KB	0 ms	1096 B

Java

```
1 import java.util.*;
2 public class Main {
3     public static void main(String args[]) {
4         Scanner sc = new Scanner(System.in);
5         int t = sc.nextInt();
6         while (t-- > 0) {
7             int n = sc.nextInt();
8             int[] from = new int[n];
9             int[] how = new int[n];
10            int[] dist = new int[n];
11            for (int i=0; i<n; i++) {
12                from[i] = how[i] = dist[i] = -1;
13            }
14            Queue<Integer> q = new LinkedList<>();
15            q.add(1%n);
16            dist[1%n] = 0;
17            how[1%n] = 1;
18            while (!q.isEmpty()) {
19                int now = q.remove();
20                for (int i=0; i<=1; i++) {
21                    int next = (now*10+i)%n;
22                    if (dist[next] == -1) {
23                        dist[next] = dist[now] + 1;
24                        from[next] = now;
25                        how[next] = i;
26                        q.add(next);
27                    }
28                }
29            }
30            if (dist[0] == -1) {
31                System.out.println("BRAK");
32            } else {
33                StringBuilder ans = new StringBuilder();
34                for (int i=0; i!=-1; i=from[i]) {
35                    ans.append(Integer.toString(how[i]));
36                }
37                System.out.println(ans.reverse());
38            }
39        }
40    }
41 }
42 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	12760 KB	128 ms	1352 B



C++14

```
1 #include <iostream>
2 #include <tuple>
3 #include <queue>
4 #include <string>
5 #include <vector>
6 using namespace std;
7 int main() {
8     int n, k;
9     cin >> n >> k;
10    vector<string> a(2);
11    for (int i=0; i<2; i++) {
12        cin >> a[i];
13    }
14    vector<pair<int,int>> dirs = {{0,1},{0,-1},{1,k}};
15    vector<vector<int>>> d(2, vector<int>(n, -1));
16    queue<pair<int,int>> q;
17    d[0][0] = 0;
18    q.push(make_pair(0,0));
19    bool ok = false;
20    while (!q.empty()) {
21        int x, y;
22        tie(x,y) = q.front(); q.pop();
23        for (auto &dir : dirs) {
24            int dx,dy;
25            tie(dx,dy) = dir;
26            int nx = (x+dx)%2;
27            int ny = y+dy;
28            if (ny >= n) {
29                ok = true;
30                break;
31            }
32            if (ny < 0) continue;
33            if (d[nx][ny] != -1) continue;
34            if (a[nx][ny] == '0') continue;
35            if (ny < d[x][y]+1) continue;
36            d[nx][ny] = d[x][y] + 1;
37            q.push(make_pair(nx,ny));
38        }
39        if (ok) break;
40    }
41    cout << (ok ? "1" : "0") << '\n';
42    return 0;
43 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	3412 KB	12 ms	1085 B

Java

```
1 import java.util.*;
2 public class Main {
3     public static void main(String args[]) {
4         Scanner sc = new Scanner(System.in);
5         int n = sc.nextInt();
6         int k = sc.nextInt();
7         String[] a = new String[2];
8         for (int i=0; i<2; i++) {
9             a[i] = sc.next();
10        }
11        int[][] dirs = {{0,1},{0,-1},{1,k}};
12        int[][] d = new int[2][n];
13        for (int i=0; i<2; i++) {
14            Arrays.fill(d[i],-1);
15        }
16        Queue<Integer> q = new LinkedList<>();
17        d[0][0] = 0;
18        q.add(0); q.add(0);
19        boolean ok = false;
20        while (!q.isEmpty()) {
21            int x = q.remove();
22            int y = q.remove();
23            for (int[] dir : dirs) {
24                int dx = dir[0];
25                int dy = dir[1];
26                int nx = (x+dx)%2;
27                int ny = y+dy;
28                if (ny >= n) {
29                    ok = true;
30                    break;
31                }
32                if (ny < 0) continue;
33                if (d[nx][ny] != -1) continue;
34                if (a[nx].charAt(ny) == '0') continue;
35                if (ny < d[x][y]+1) continue;
36                d[nx][ny] = d[x][y] + 1;
37                q.add(nx); q.add(ny);
38            }
39            if (ok) break;
40        }
41        System.out.println(ok ? "1" : "0");
42    }
43 }
44 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	31404 KB	372 ms	1321 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](mailto:codeplus@startlink.io) 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.