

专题1

目录

Problem ID	Title
1001	Nightmare
1002	连连看
1003	诡异的楼梯
1004	变形课
1005	Rescue
1006	胜利大逃亡
1007	非常可乐
1008	逃离迷宫
1009	推箱子
1010	胜利大逃亡(续)

1001 Nightmare

AC代码

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int t;
4  int startx, starty, endx, endy;
5  int Map[11][11];
6  int cge[4][2] = { {0,1},{0,-1},{-1,0},{1,0} };
7  struct pos {
8      int x, y;
9      int step;
10     int time;
11 };
12 int n, m;
13 void bfs();
14 int main()
15 {
16     scanf("%d", &t);
17     for (int p = 1; p <= t; p++)
18     {
19         scanf("%d%d", &n, &m);
20         for (int i = 1; i <= n; i++)
21             for (int j = 1; j <= m; j++) {
22                 scanf("%d", &Map[i][j]);
```

```

23         if (Map[i][j] == 2) {
24             startx = i; starty = j;
25         }
26         if (Map[i][j] == 3) {
27             endx = i; endy = j;
28         }
29     }
30     bfs();
31 }
32 return 0;
33 }
34 void bfs()
35 {
36     pos cur, nex;
37     cur.x = startx;
38     cur.y = starty;
39     cur.step = 0;
40     cur.time = 6;
41     queue<pos>qu;
42     qu.push(cur);
43     while (!qu.empty()) {
44         cur = qu.front();
45         qu.pop();
46         if (cur.x == endx && cur.y == endy && cur.time >= 1) {
47             printf("%d\n", cur.step);
48             return;
49         }
50         for (int i = 0; i < 4; i++) {
51             nex.x = cur.x + cge[i][0];
52             nex.y = cur.y + cge[i][1];
53             nex.step = cur.step + 1;
54             nex.time = cur.time - 1;
55             if (Map[nex.x][nex.y] == 4 && cur.time >= 2) nex.time = 6;
56             if (nex.x <= n && nex.x >= 1 && nex.y >= 1 && nex.y <= m &&
nex.time >= 1 && Map[nex.x][nex.y] != 0) {
57                 if (Map[nex.x][nex.y] == 4) Map[nex.x][nex.y] = 0;
58                 qu.push(nex);
59             }
60         }
61     }
62     printf("-1\n");
63     return;
64 }

```

1002 连连看

AC代码

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  int n, m;
4  int sx, sy, ex, ey;
5  int Map[1010][1010];
6  int vis[1010][1010];

```

```

7  int q;
8  int cge[4][2] = { {0,1},{0,-1},{-1,0},{1,0} };
9  struct pos {
10     int x, y;
11     int tx, ty; /*上次的操作*/
12     int step; /*转弯次数*/
13 };
14 void bfs();
15 int main()
16 {
17     while (scanf("%d%d", &n, &m), n && m)
18     {
19         for (int i = 1; i <= n; i++)
20             for (int j = 1; j <= m; j++) {
21                 scanf("%d", &Map[i][j]);
22             }
23         scanf("%d", &q);
24         for (int i = 1; i <= q; i++) {
25             scanf("%d%d%d%d", &sx, &sy, &ex, &ey);
26             int t = Map[ex][ey];
27             bfs();
28             Map[ex][ey] = t;
29         }
30     }
31     return 0;
32 }
33 void bfs()
34 {
35     if (Map[sx][sy] != Map[ex][ey]) {
36         printf("NO\n");
37         return;
38     }
39     if (Map[sx][sy] == 0 || Map[ex][ey] == 0) {
40         printf("NO\n");
41         return;
42     }
43     Map[ex][ey] = 0;
44     memset(vis, 0, sizeof(vis));
45     vis[sx][sy] = 1;
46     pos cur, nex;
47     cur.x = sx;
48     cur.y = sy;
49     cur.tx = 0;
50     cur.ty = 0;
51     cur.step = 0;
52     queue<pos>qu;
53     qu.push(cur);
54     while (!qu.empty()) {
55         cur = qu.front();
56         qu.pop();
57         if (cur.x == ex && cur.y == ey && cur.step <= 3) {
58             printf("YES\n");
59             return;
60         }
61         for (int i = 0; i < 4; i++) {
62             nex.x = cur.x + cge[i][0];
63             nex.y = cur.y + cge[i][1];
64             nex.tx = cge[i][0];

```

```

65         nex.ty = cge[i][1];
66         if ((nex.tx != cur.tx) || (nex.ty != cur.ty))nex.step = cur.step
+ 1;
67         else nex.step = cur.step;
68         if (nex.x <= n && nex.x >= 1 && nex.y <= m && nex.y >= 1 ){
69             if(nex.step <= 3 && Map[nex.x][nex.y] == 0 && vis[nex.x]
[nex.y] == 0) {
70                 vis[nex.x][nex.y] = 1;
71                 qu.push(nex);
72             }
73         }
74     }
75 }
76 printf("NO\n");
77 return;
78 }
79

```

1003 诡异的楼梯

AC代码

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  int m, n;
4  int sx, sy, ex, ey;
5  char Map[25][25];
6  int vis[25][25];
7  int dir[4][2] = { {0,1},{0,-1},{1,0},{-1,0} };
8  struct pos {
9      int x, y;
10     int time;
11 };
12 void bfs();
13
14 int main()
15 {
16     while (scanf("%d%d", &m, &n) != EOF) {
17         for (int i = 1; i <= m; i++)
18             {
19                 for (int j = 1; j <= n; j++)
20                     {
21                         cin >> Map[i][j];
22                         vis[i][j] = 0;
23                         if (Map[i][j] == 'S') {
24                             sx = i, sy = j;
25                             vis[i][j] = 1;
26                         }
27                         if (Map[i][j] == 'T') {
28                             ex = i, ey = j;
29                         }
30                     }
31             }
32     Map[ex][ey] = '.'; /*便于判断移动条件*/

```

```

33     bfs();
34 }
35     return 0;
36 }
37 void bfs()
38 {
39     pos cur, nex;
40     cur.x = sx, cur.y = sy;
41     cur.time = 0;
42     queue<pos>qu;
43     qu.push(cur);
44     while (!qu.empty()) {
45         cur = qu.front();
46         qu.pop();
47         if (cur.x == ex && cur.y == ey) {
48             printf("%d\n", cur.time);
49             return;
50         }
51         for (int i = 0; i < 4; i++) {
52             nex.x = cur.x + dir[i][0];
53             nex.y = cur.y + dir[i][1];
54             nex.time = cur.time + 1;
55             if (nex.x <= m && nex.x >= 1 && nex.y <= n && nex.y >= 1 &&
vis[nex.x][nex.y] == 0) {
56                 if (Map[nex.x][nex.y] == '|') {
57                     if (((i == 0 || i == 1) && cur.time % 2 == 1) || ((i ==
2 || i == 3) && cur.time % 2 == 0)) {
58                         if (Map[nex.x + dir[i][0]][nex.y + dir[i][1]] ==
'.'&& vis[nex.x + dir[i][0]][nex.y + dir[i][1]]==0) {
59                             vis[nex.x + dir[i][0]][nex.y + dir[i][1]] = 1;
60                             nex.x += dir[i][0];
61                             nex.y += dir[i][1];
62                             qu.push(nex);
63                         }
64                     }
65                     else if (Map[nex.x + dir[i][0]][nex.y + dir[i][1]] ==
'.'.') {
66                         nex.x -= dir[i][0];
67                         nex.y -= dir[i][1];
68                         qu.push(nex);
69                     }
70                     else continue;
71                 }
72                 else if (Map[nex.x][nex.y] == '-') {
73                     if (((i == 2 || i == 3) && cur.time % 2 == 1) || ((i ==
0 || i == 1) && cur.time % 2 == 0)) {
74                         if (Map[nex.x + dir[i][0]][nex.y + dir[i][1]] ==
'.'&& vis[nex.x + dir[i][0]][nex.y + dir[i][1]]==0){
75                             vis[nex.x + dir[i][0]][nex.y + dir[i][1]] = 1;
76                             nex.x += dir[i][0];
77                             nex.y += dir[i][1];
78                             qu.push(nex);
79                         }
80                     }
81                     else if (Map[nex.x + dir[i][0]][nex.y + dir[i][1]] ==
'.'.') {
82                         nex.x -= dir[i][0];
83                         nex.y -= dir[i][1];

```

```

84         qu.push(nex);
85     }
86     else continue;
87 }
88 else if (Map[nex.x][nex.y] == '.') {
89     vis[nex.x][nex.y] = 1;
90     qu.push(nex);
91 }
92 }
93 }
94 }
95 }
96

```

1004 变形课

AC代码

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  struct turn {
4      char fr;
5      char to;
6      int flag;
7  }s[10000];
8  char a[10000];
9  int cnt;
10 void bfs();
11 int main()
12 {
13     while (scanf("%s", a) != EOF) {
14         int len = strlen(a);
15         cnt = 0;
16         s[++cnt].fr = a[0];
17         s[cnt].to = a[len - 1];
18         s[cnt].flag = 0;
19         while (++cnt) {
20             scanf("%s", a);
21             if (a[0] == '0')break;
22             len = strlen(a);
23             s[cnt].fr = a[0];
24             s[cnt].to = a[len - 1];
25             s[cnt].flag = 0;
26         }
27         bfs();
28     }
29     return 0;
30 }
31 void bfs()
32 {
33     queue<turn>qu;
34     turn cur, nex;
35     cur.to = 'b';
36     cur.flag = 1;

```

```

37     qu.push(cur);
38     while (!qu.empty()) {
39         cur = qu.front();
40         qu.pop();
41         if (cur.to == 'm') {
42             printf("Yes.\n");
43             return;
44         }
45         for (int i = 1; i <= cnt - 1; i++) {
46             if (s[i].fr == cur.to && s[i].flag == 0) {
47                 nex = s[i];
48                 s[i].flag = 1;
49                 qu.push(nex);
50             }
51         }
52     }
53     printf("No.\n");
54 }

```

1005 Rescue

AC代码

```

1 #include<bits/stdc++.h>
2 using namespace std;
3 char Map[210][210];
4 int sx[210], sy[210], ex, ey;
5 int n, m;
6 int cnt = 0; /*朋友数*/
7 struct pos {
8     int x;
9     int y;
10    int time;
11    int f;
12 };
13 int vis[210][210];
14 int dir[4][2] = { {-1,0},{1,0},{0,1},{0,-1} };
15 void bfs();
16 int main()
17 {
18     while (scanf("%d%d", &n, &m)!=EOF) {
19         for (int i = 1; i <= n; i++)
20             for (int j = 1; j <= m; j++) {
21                 cin >> Map[i][j];
22                 vis[i][j] = 0;
23                 if (Map[i][j] == 'r') {
24                     cnt++;
25                     sx[cnt] = i;
26                     sy[cnt] = j;
27                 }
28                 if (Map[i][j] == 'a') {
29                     ex = i, ey = j;
30                     vis[i][j] = 1;
31                 }
32             }
33     }
34 }

```

```

32     }
33     bfs();
34 }
35 return 0;
36 }
37 void bfs()
38 {
39     queue<pos>qu;
40     pos cur, nex;
41     cur.x = ex, cur.y = ey;
42     cur.time = 0;
43     cur.f = 0;
44     qu.push(cur);
45     while (!qu.empty()) {
46         cur = qu.front();
47         qu.pop();
48         if (Map[cur.x][cur.y] == 'r') {
49             printf("%d\n", cur.time);
50             return;
51         }
52         nex.time = cur.time + 1;
53         if (cur.f) {
54             nex.x = cur.x, nex.y = cur.y;
55             nex.f = 0;
56             qu.push(nex);
57             continue;
58         }
59         nex.f = 0;
60         for (int i = 0; i < 4; i++) {
61             if (Map[cur.x + dir[i][0]][cur.y + dir[i][1]] != '#' &&
vis[cur.x + dir[i][0]][cur.y + dir[i][1]] == 0) {
62                 if (cur.x + dir[i][0] >= 1 && cur.x + dir[i][0] <= n &&
cur.y + dir[i][1] >= 1 && cur.y + dir[i][1] <= m) {
63                     nex.x = cur.x + dir[i][0];
64                     nex.y = cur.y + dir[i][1];
65                     vis[nex.x][nex.y] = 1;
66                     if (Map[nex.x][nex.y] == 'x')nex.f = 1;
67                     qu.push(nex);
68                 }
69             }
70         }
71     }
72     printf("Poor ANGEL has to stay in the prison all his life.\n");
73 }
74

```

1006 胜利大逃亡

AC代码

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int Map[60][60][60];
4  int vis[60][60][60];
5  int a, b, c, t;
6  int dir[6][3] = { {0,0,1},{0,0,-1}, {0,1,0}, {0,-1,0}, {1,0,0}, {-1,0,0} };
7  struct pos {
8      int x, y, z;
9      int time;
10 };
11 int q;
12 void bfs();
13 int main()
14 {
15     scanf("%d", &q);
16     for (int p = 1; p <= q; p++) {
17         scanf("%d%d%d", &a, &b, &c, &t);
18         for(int i=1;i<=a;i++)
19             for(int j=1;j<=b;j++)
20                 for (int k = 1; k <= c; k++) {
21                     scanf("%d", &Map[i][j][k]);
22                     vis[i][j][k] = 0;
23                 }
24         bfs();
25     }
26     return 0;
27 }
28 void bfs()
29 {
30     if (Map[a][b][c] == 1) {
31         printf("-1\n");
32         return;
33     }
34     queue<pos>qu;
35     pos cur, nex;
36     cur.time = 0;
37     cur.x = 1, cur.y = 1, cur.z = 1;
38     qu.push(cur);
39     vis[1][1][1] = 1;
40     while (!qu.empty()) {
41         cur = qu.front();
42         qu.pop();
43         if (cur.x == a && cur.y == b && cur.z == c) {
44             printf("%d\n", cur.time);
45             return;
46         }
47         if (cur.time >= t) {
48             printf("-1\n");
49             return;
50         }
51         for (int i = 0; i < 6; i++) {
52             nex.time = cur.time + 1;
53             nex.x = cur.x + dir[i][0];
54             nex.y = cur.y + dir[i][1];
55             nex.z = cur.z + dir[i][2];
```

```

56         if (Map[nex.x][nex.y][nex.z] == 1)continue;
57         if (nex.x<1 || nex.x>a || nex.y<1 || nex.y>b || nex.z<1 ||
nex.z>c)continue;
58         if (vis[nex.x][nex.y][nex.z])continue;
59         vis[nex.x][nex.y][nex.z] = 1;
60         qu.push(nex);
61     }
62 }
63 printf("-1\n");
64 }

```

1007 非常可乐

AC代码

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  int s, n, m;
4  struct co {
5      int s, n, m;
6      int step;
7  };
8  int vis[105][105][105];
9  void bfs();
10 int main()
11 {
12     while (scanf("%d%d%d", &s, &n, &m)) {
13         if (s == 0 && n == 0 && m == 0)break;
14         bfs();
15         memset(vis, 0, sizeof(vis));
16     }
17     return 0;
18 }
19 void bfs()
20 {
21     queue<co>qu;
22     co cur, nex;
23     cur.s = s, cur.n = 0, cur.m = 0;
24     cur.step = 0;
25     qu.push(cur);
26     vis[s][0][0] = 1;
27     while (!qu.empty()) {
28         cur = qu.front();
29         qu.pop();
30         if ((cur.s == cur.n && cur.m == 0) || (cur.s == cur.m && cur.n ==
0) || (cur.m == cur.n && cur.s == 0)) {
31             printf("%d\n", cur.step);
32             return;
33         }
34         for (int i = 1; i <= 6; i++)/*六种倒水方式*/
35         {
36             if (i == 1) {
37                 if (cur.s >= n - cur.n) {
38                     nex.s = cur.s - n + cur.n;

```

```

39         nex.n = n;
40         nex.m = cur.m;
41     }
42     else {
43         nex.s = 0;
44         nex.n = cur.n + cur.s;
45         nex.m = cur.m;
46     }
47 }
48 else if (i == 2) {
49     if (cur.s >= m - cur.m) {
50         nex.s = cur.s - m + cur.m;
51         nex.m = m;
52         nex.n = cur.n;
53     }
54     else {
55         nex.s = 0;
56         nex.m = cur.m + cur.s;
57         nex.n = cur.n;
58     }
59 }
60 else if (i == 3) {
61     if (cur.m >= n - cur.n) {
62         nex.m = cur.m - n + cur.n;
63         nex.n = n;
64         nex.s = cur.s;
65     }
66     else {
67         nex.m = 0;
68         nex.n = cur.n + cur.m;
69         nex.s = cur.s;
70     }
71 }
72 else if (i == 4) {
73     if (cur.n >= m - cur.m) {
74         nex.n = cur.n - m + cur.m;
75         nex.m = m;
76         nex.s = cur.s;
77     }
78     else {
79         nex.n = 0;
80         nex.m = cur.m + cur.n;
81         nex.s = cur.s;
82     }
83 }
84 else if (i == 5) {
85     if (cur.n >= s - cur.s) {
86         nex.n = cur.n - s + cur.s;
87         nex.s = s;
88         nex.m = cur.m;
89     }
90     else {
91         nex.n = 0;
92         nex.s = cur.s + cur.n;
93         nex.m = cur.m;
94     }
95 }
96 else if (i == 6) {

```

```

97         if (cur.m >= s - cur.s) {
98             nex.m = cur.m - s + cur.s;
99             nex.s = s;
100             nex.n = cur.n;
101         }
102         else {
103             nex.m = 0;
104             nex.s = cur.m + cur.s;
105             nex.n = cur.n;
106         }
107     }
108     if (vis[nex.s][nex.n][nex.m] == 0) {
109         vis[nex.s][nex.n][nex.m] = 1;
110         nex.step = cur.step + 1;
111         qu.push(nex);
112     }
113 }
114 }
115 printf("NO\n");
116 }

```

1008 逃离迷宫

AC代码

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  int m, n; /*m行n列*/
4  int t; /*测试组数*/
5  int k; /*最大转弯次数*/
6  char Map[110][110];
7  int sx, sy, ex, ey; /*起点终点坐标*/
8  struct pos {
9      int x, y;
10     int step;
11 };
12 int vis[110][110]; /*是否入过队*/
13 int dir[4][2] = { {-1,0},{1,0},{0,1},{0,-1} }; /*方向*/
14 void bfs();
15 int main()
16 {
17     scanf("%d", &t);
18     for (int p = 1; p <= t; p++) {
19         scanf("%d%d", &m, &n);
20         for(int i=1;i<=m;i++)
21             for (int j = 1; j <= n; j++) {
22                 cin >> Map[i][j];
23                 vis[i][j] = 0;
24             }
25         scanf("%d%d%d%d", &k, &sy, &sx, &ey, &ex); /*题目先给列再给行*/
26         bfs();
27     }
28     return 0;
29 }

```

```

30 void bfs()
31 {
32     queue<pos>qu;
33     pos cur, nex;
34     cur.step = -1;
35     cur.x = sx, cur.y = sy;
36     qu.push(cur);
37     vis[sx][sy] = 1;
38     while (!qu.empty()) {
39         cur = qu.front();
40         qu.pop();
41         if (cur.x == ex && cur.y == ey && cur.step <= k) {
42             printf("yes\n");
43             return;
44         }
45         for (int i = 0; i < 4; i++) {
46             nex.x = cur.x + dir[i][0];
47             nex.y = cur.y + dir[i][1];
48             while (nex.x <= m && nex.x >= 1 && nex.y <= n && nex.y >= 1
&&Map[nex.x][nex.y]=='.') {
49                 if (vis[nex.x][nex.y] == 0) {
50                     vis[nex.x][nex.y] = 1;
51                     nex.step = cur.step + 1;
52                     qu.push(nex);
53                 }
54                 nex.x += dir[i][0];
55                 nex.y += dir[i][1];
56             }
57         }
58     }
59     printf("no\n");
60 }

```

1009 推箱子

未AC

1010 胜利大逃亡（续）

未AC
