



지배권

acmicpc.net

21

그 2/4 > 2019년 8월 17일

Psh

ShiftPsh



22 23

10-6

922

10-111 (5m)  
↓ 122 (2-3)

211302

224

11-21

12m

(34)

60

61

12m

## C++14

```

1 #include <iostream>
2 #include <algorithm>
3 #include <vector>
4 using namespace std;
5 int main() {
6     int h, w;
7     cin >> h >> w;
8     int n;
9     cin >> n;
10    vector<int> r(n), c(n);
11    for (int i=0; i<n; i++) {
12        cin >> r[i] >> c[i];
13    }
14    int ans = 0;
15    for (int i=0; i<n; i++) {
16        int r1 = r[i], c1 = c[i];
17        for (int j=i+1; j<n; j++) {
18            int r2 = r[j], c2 = c[j];
19            for (int rot1=0; rot1<2; rot1++) {
20                for (int rot2=0; rot2<2; rot2++) {
21                    if (r1+r2 <= h && max(c1,c2) <= w) {
22                        int temp = r1*c1+r2*c2;
23                        if (ans < temp) ans = temp;
24                    }
25                    if (max(r1,r2) <= h && c1+c2 <= w) {
26                        int temp = r1*c1+r2*c2;
27                        if (ans < temp) ans = temp;
28                    }
29                    swap(r2, c2);
30                }
31                swap(r1, c1);
32            }
33        }
34    }
35    cout << ans << '\n';
36    return 0;
37 }
38

```

결과

메모리

시간

코드 길이

맞았습니다!!

1988 KB

0 ms

1008 B

C++14

```
1 #include <iostream>
2 #include <tuple>
3 #include <vector>
4 #include <cstring>
5 using namespace std;
6 int a[100][100];
7 int n;
8 int go(int x, int y, int dir) {
9     if (x == n-1 && y == n-1) {
10         return 1;
11     }
12     int ans = 0;
13     if (dir == 0) { // -
14         if (y+1 < n && a[x][y+1] == 0) {
15             ans += go(x, y+1, 0);
16         }
17         if (x+1 < n && y+1 < n && a[x][y+1] == 0 && a[x+1][y] == 0 && a[x+1][y+1] == 0) {
18             ans += go(x+1, y+1, 1);
19         }
20     } else if (dir == 1) { // dig
21         if (y+1 < n && a[x][y+1] == 0) {
22             ans += go(x, y+1, 0);
23         }
24         if (x+1 < n && a[x+1][y] == 0) {
25             ans += go(x+1, y, 2);
26         }
27         if (x+1 < n && y+1 < n && a[x][y+1] == 0 && a[x+1][y] == 0 && a[x+1][y+1] == 0) {
28             ans += go(x+1, y+1, 1);
29         }
30     } else if (dir == 2) { // |
31         if (x+1 < n && a[x+1][y] == 0) {
32             ans += go(x+1, y, 2);
33         }
34         if (x+1 < n && y+1 < n && a[x][y+1] == 0 && a[x+1][y] == 0 && a[x+1][y+1] == 0) {
35             ans += go(x+1, y+1, 1);
36         }
37     }
38     return ans;
39 }
40 int main() {
41     cin >> n;
42     for (int i=0; i<n; i++) {
43         for (int j=0; j<n; j++) {
44             cin >> a[i][j];
45         }
46     }
47     cout << go(0, 1, 0) << '\n';
48     return 0;
49 }
50
```

결과

메모리

시간

코드 길이

맞았습니다!!

2028 KB

24 ms

1271 B

## C++14

```

1 #include <iostream>
2 using namespace std;
3 int n, m;
4 string a[20];
5 int dx[] = {0,0,1,-1};
6 int dy[] = {1,-1,0,0};
7 int go(int step, int x1, int y1, int x2, int y2) {
8     if (step == 11) return -1;
9     bool fall1=false, fall2=false;
10    if (x1 < 0 || x1 >= n || y1 < 0 || y1 >= m) fall1 = true;
11    if (x2 < 0 || x2 >= n || y2 < 0 || y2 >= m) fall2 = true;
12    if (fall1 && fall2) return -1;
13    if (fall1 || fall2) return step;
14    int ans = -1;
15    for (int k=0; k<4; k++) {
16        int nx1 = x1+dx[k];
17        int ny1 = y1+dy[k];
18        int nx2 = x2+dx[k];
19        int ny2 = y2+dy[k];
20        if (0 <= nx1 && nx1 < n && 0 <= ny1 && ny1 < m && a[nx1][ny1] == '#') {
21            nx1 = x1;
22            ny1 = y1;
23        }
24        if (0 <= nx2 && nx2 < n && 0 <= ny2 && ny2 < m && a[nx2][ny2] == '#') {
25            nx2 = x2;
26            ny2 = y2;
27        }
28
29        int temp = go(step+1, nx1, ny1, nx2, ny2);
30        if (temp == -1) continue;
31        if (ans == -1 || ans > temp) {
32            ans = temp;
33        }
34    }
35    return ans;
36 }
37 int main() {
38     cin >> n >> m;
39     int x1,y1,x2,y2;
40     x1=y1=x2=y2=-1;
41     for (int i=0; i<n; i++) {
42         cin >> a[i];
43         for (int j=0; j<m; j++) {
44             if (a[i][j] == 'o') {
45                 if (x1 == -1) {
46                     x1 = i;
47                     y1 = j;
48                 } else {
49                     x2 = i;
50                     y2 = j;
51                 }
52                 a[i][j] = '.';
53             }
54         }
55     }
56     cout << go(0,x1,y1,x2,y2) << '\n';
57     return 0;
58 }
59

```



C++14

```
1 #include <iostream>
2 #include <algorithm>
3 #include <vector>
4 #include <tuple>
5 using namespace std;
6 void go(vector<vector<int>> &a, tuple<int,int,int> t) {
7     int row, col, size;
8     tie(row, col, size) = t;
9     vector<vector<int>> groups;
10    for (int s=1; s<=size; s++) {
11        vector<int> group;
12        // (r-s, c-s) -> (r-s, c+s)
13        for (int r=row-s, c=col-s; c<col+s; c++) {
14            group.push_back(a[r][c]);
15        }
16        // (r-s, c+s) -> (r+s, c+s)
17        for (int r=row-s, c=col+s; r<row+s; r++) {
18            group.push_back(a[r][c]);
19        }
20        // (r+s, c+s) -> (r+s, c-s)
21        for (int r=row+s, c=col+s; c>col-s; c--) {
22            group.push_back(a[r][c]);
23        }
24        // (r+s, c-s) -> (r-s, c-s)
25        for (int r=row+s, c=col-s; r>row-s; r--) {
26            group.push_back(a[r][c]);
27        }
28        groups.push_back(group);
29    }
30    for (int s=1; s<=size; s++) {
31        auto &group = groups[s-1];
32        rotate(group.rbegin(), group.rbegin()+1, group.rend());
33        int len = group.size();
34        int index = 0;
35        // (r-s, c-s) -> (r-s, c+s)
36        for (int r=row-s, c=col-s; c<col+s; c++) {
37            a[r][c] = group[index++];
38        }
39        // (r-s, c+s) -> (r+s, c+s)
40        for (int r=row-s, c=col+s; r<row+s; r++) {
41            a[r][c] = group[index++];
42        }
43        // (r+s, c+s) -> (r+s, c-s)
44        for (int r=row+s, c=col+s; c>col-s; c--) {
45            a[r][c] = group[index++];
46        }
47        // (r+s, c-s) -> (r-s, c-s)
48        for (int r=row+s, c=col-s; r>row-s; r--) {
49            a[r][c] = group[index++];
50        }
51    }
52 }
53 int main() {
54     int n, m, k;
55     cin >> n >> m >> k;
56     vector<vector<int>> a(n, vector<int>(m));
57     for (int i=0; i<n; i++) {
58         for (int j=0; j<m; j++) {
59             cin >> a[i][j];
60         }
61     }
62     vector<tuple<int, int, int>> d(k);
63     for (int i=0; i<k; i++) {
64         int r, c, s;
65         cin >> r >> c >> s;
66         d[i] = make_tuple(r-1, c-1, s);
67     }
68     sort(d.begin(), d.end());
69     int ans = 100*100;
70     do {
71         auto b = a;
72         for (auto &t : d) {
73             go(b, t);
74         }
75         for (int i=0; i<n; i++) {
76             int sum = 0;
77             for (int j=0; j<m; j++) {
78                 sum += b[i][j];
79             }
80             if (ans > sum) ans = sum;
81         }
82     } while (next_permutation(d.begin(), d.end()));
83     cout << ans << '\n';
84     return 0;
85 }
```



$2z > \rho'_2$

$\frac{5}{2}$

백준이는 비서에게 최대한 많은 상담을 잡으라고 부탁을 했고, 비서는 하루에 하나씩 서로 다른 사람의 상담을 잡아놓았다.

$N+1$ 일

각각의 상담은 상담을 완료하는데 걸리는 시간  $T_i$ 와 상담을 했을 때 받을 수 있는 금액  $P_i$ 로 이루어져 있다

$N = 7$ 인 경우에 다음과 같은 상담 일정표를 보자.

	1일	2일	3일	4일	5일	6일	7일
$T_i$	3	5	1	1	2	4	2
$P_i$	10	20	10	20	15	40	200

↓

154

1일에 잡혀있는 상담은 총 3일이 걸리며, 상담했을 때 받을 수 있는 금액은 10이다. 5일에 잡혀있는 상담은 총 2일이 걸리며, 받을 수 있는 금액은 15이다.

상담을 하는데 필요한 기간은 1일보다 클 수 있기 때문에, 모든 상담을 할 수는 없다. 예를 들어서 1일에

7월 3

$1\frac{1}{2}$	$2\frac{1}{2}$	$3\frac{1}{2}$	$4\frac{1}{2}$	$5\frac{1}{2}$		

1월 1일 :

수입 : 80원

1월 2일 :

수입 : 100원

1월 3일 :

수입 : 70원
