

- All
- Mine
- Followed

```
61         return m.minZ() || x < m[y].size();
62         return 0;
63     unordered_map<MemoKey, int, MemoKeyHash> mZ;
64     int zerosC = zeros(x, y, !s, m, mZ);
65     int minZ = (minZeros != 0)
66                 ? min(minZeros, zerosC)
67                 : zerosC;
68     MemoKey key = {x, y, minZ, s};
69     if (memo.find(key) != memo.end())
70     {
71         return memo[key];
72     }
73     int maxA = 0;
74     if (minZ == 0)
75     {
76         maxA = maxArea((s ? x + 1 : x),
77                        (s ? y : y + 1), s, 1, minZ, m, memo);
78     }
79     else
80     {
81         maxA = max(minZ * k, maxArea((s ? x + 1 : x),
82                                     (s ? y : y + 1), s, k + 1, minZ, m, memo));
83     }
84     memo[key] = maxA;
85     return maxA;
86 }
87
88 int max_land(int x, int y, bool s,
89             const vector<vector<int>> &m)
90 {
91     int r = 0;
92     unordered_map<MemoKey, int, MemoKeyHash> memo;
93
94     if (s && x >= 0 && y >= 0)
95     {
96         for (int i = x; i < m[y].size(); ++i)
97         {
98             r = max(r, maxArea(i, y, s, 1, 0, m, memo));
99         }
100        if (x == 0)
101        {
102            r = max(r, max_land(x, y - 1, s, m));
103        }
104        else
105        {
106            r = max(r, max_land(x - 1, y, !s, m));
107        }
108    }
109    else if (!s && y >= 0 && x >= 0)
110    {
111        for (int i = y; i < m.size(); ++i)
112        {
113            r = max(r, maxArea(x, i, s, 1, 0, m, memo));
114        }
115        if (y == 0)
116        {
117            r = max(r, max_land(x - 1, y, s, m));
118        }
119        else
120        {
121            r = max(r, max_land(x, y - 1, !s, m));
122        }
123    }
124    return r;
125 }
126 int main()
127 {
128     int m, n;
129     cin >> m >> n;
130     while (m != 0 || n != 0)
131     {
132         vector<vector<int>> grid(m, vector<int>(n));
133
134         for (int i = 0; i < m; ++i)
```

mit Time

min ago

hr ago

hr ago

hr ago