

```

1  #include <bits/stdc++.h>
2  #define ll long long
3
4  using namespace std;
5
6  const int MOD = 1e9 + 7;
7
8  struct Binomial {
9      int mxn = -1;
10     vector<int> inv, fact;
11
12     // inverse = pow(a, mod - 2);
13     int inverse(int a) {
14         int m = MOD;
15         int u = 0, v = 1;
16         while (a != 0) {
17             int t = m / a;
18             m -= t * a;
19             swap(a, m);
20             u -= t * v;
21             swap(u, v);
22         }
23         assert(m == 1);
24         return (u + MOD) % MOD;
25     }
26
27     void preprocess() {
28         fact[0] = 1;
29         for (int i = 1; i < mxn; i++) {
30             fact[i] = (ll)fact[i-1] * i % MOD;
31         }
32     }
33
34     Binomial(int mxn = (int)2e5 + 10) {
35         this->mxn = mxn;
36         fact.resize(mxn);
37         preprocess();
38     }
39
40     int C(int n, int k) {
41         if (k > n) swap(k, n);
42         return ((ll)fact[n] * inverse((ll)fact[k] * fact[n-k] % MOD)) % MOD;
43     }
44 };
45
46 int main() {
47     int w, h, n;
48     cin >> w >> h >> n;
49
50     n += 2;
51
52     vector<pair<int, int>> points(n);
53     for (int i = 1; i < n-1; i++) {
54         int x, y;
55         cin >> x >> y;
56         x--; y--;
57         points[i] = { x, y };
58     }
59
60     points[0] = { 0, 0 };
61     points[n-1] = { w - 1, h - 1 };
62
63     sort(points.begin(), points.end());
64
65     Binomial b(2e5 + 10);
66     vector<ll> f(n), g(n);
67     g[0] = 0;
68
69     for (int i = 1; i < n; i++) {
70         int x1, y1;
71         tie(x1, y1) = points[i];
72
73         for (int j = i-1; j ≥ 0; j--) {
74             int x2, y2;
75             tie(x2, y2) = points[j];
76
77             if (y2 > y1) continue;
78
79             ll X = x1 - x2;
80             ll Y = y1 - y2;
81
82             f[i] = (f[i] + (g[j] * b.C(X + Y, Y)) % MOD) % MOD;
83         }
84
85         g[i] = (b.C(x1 + y1, y1) - f[i] + MOD) % MOD;
86     }
87
88     cout << g[n-1] << endl;
89     return 0;
90 }

```