|  |
| --- |
| #include <algorithm> |
|  | #include <cstdio> |
|  | #include <cstdlib> |
|  | #include <iostream> |
|  | #include <map> |
|  | #include <tuple> |
|  | #include <vector> |
|  | using namespace std; |
|  |  |
|  | int main() { |
|  | int N, X, Y; |
|  | while (cin >> N >> X >> Y) { |
|  | vector<tuple<int, int, int, long long>> v; |
|  | long long shash = 0; |
|  | for (int i = 0; i < N; i++) { |
|  | int x1, y1, x2, y2; |
|  | cin >> x1 >> y1 >> x2 >> y2; |
|  | int d1 = (x1 == 0) ? y1 : (y1 == Y) ? Y+x1 : (x1 == X) ? Y+X+Y-y1 : Y+X+Y+X-x1; |
|  | int d2 = (x2 == 0) ? y2 : (y2 == Y) ? Y+x2 : (x2 == X) ? Y+X+Y-y2 : Y+X+Y+X-x2; |
|  | long long hash = ((long long)rand()<<40)+((long long)rand()<<20)+rand(); |
|  | if (d2 < d1) shash += hash; else shash -= hash; |
|  | v.emplace\_back(d1, x1, y1, 2\*hash); |
|  | v.emplace\_back(d2, x2, y2, -2\*hash); |
|  | } |
|  | sort(v.begin(), v.end()); |
|  |  |
|  | map<long long, int> hashes; |
|  | for (int i = 0; i < v.size(); i++) { |
|  | shash += get<3>(v[i]); |
|  | hashes[shash] = i; |
|  | } |
|  | for (int i = 0; i < v.size(); i++) { |
|  | shash += get<3>(v[i]); |
|  | if (hashes.count(-shash)) { |
|  | int x1 = get<1>(v[i]), y1 = get<2>(v[i]); |
|  | int x2 = get<1>(v[hashes[-shash]]), y2 = get<2>(v[hashes[-shash]]); |
|  | printf("1\n"); |
|  | printf("%.1lf ", x1 + ((y1 == Y) ? 0.5 : (y1 == 0) ? -0.5 : 0)); |
|  | printf("%.1lf ", y1 + ((x1 == 0) ? 0.5 : (x1 == X) ? -0.5 : 0)); |
|  | printf("%.1lf ", x2 + ((y2 == Y) ? 0.5 : (y2 == 0) ? -0.5 : 0)); |
|  | printf("%.1lf\n", y2 + ((x2 == 0) ? 0.5 : (x2 == X) ? -0.5 : 0)); |
|  | goto done; |
|  | } |
|  | } |
|  | printf("2\n"); |
|  | printf("%.1lf %.1lf %.1lf %.1lf\n", 0.5, 0.0, X-0.5, (double)Y); |
|  | printf("%.1lf %.1lf %.1lf %.1lf\n", X-0.5, 0.0, 0.5, (double)Y); |
|  | done:; |
|  | } |
|  | } |