

기하 알고리즘 1

소스코드

최백준 [choi@startlink.io](mailto:choi@startlink.io)



C++14

```
1 #include <iostream>
2 using namespace std;
3 int ccw(int x1, int y1, int x2, int y2, int x3, int y3) {
4     int temp = x1*y2+x2*y3+x3*y1;
5     temp = temp - y1*x2-y2*x3-y3*x1;
6     if (temp > 0) return 1;
7     else if (temp < 0) return -1;
8     else return 0;
9 }
10 int main() {
11     int x1, y1, x2, y2, x3, y3;
12     cin >> x1 >> y1 >> x2 >> y2 >> x3 >> y3;
13     cout << ccw(x1, y1, x2, y2, x3, y3) << '\n';
14     return 0;
15 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1984 KB	0 ms	407 B

C++14

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4 using Point = pair<long long, long long>;
5 using Line = pair<Point, Point>;
6 int ccw(Point p1, Point p2, Point p3) {
7     long long temp = p1.first * p2.second + p2.first * p3.second + p3.first * p1.second;
8     temp -= p1.second * p2.first + p2.second * p3.first + p3.second * p1.first;
9     if (temp > 0) return 1;
10    if (temp < 0) return -1;
11    return 0;
12 }
13 bool cross(Line l1, Line l2) {
14     int l1l2 = ccw(l1.first, l1.second, l2.first) * ccw(l1.first, l1.second, l2.second);
15     int l2l1 = ccw(l2.first, l2.second, l1.first) * ccw(l2.first, l2.second, l1.second);
16     return l1l2 < 0 && l2l1 < 0;
17 }
18 int go(Point &p1, int n, vector<Point> &a) {
19     for (int i=0; i<n; i++) {
20         int t1 = ccw(a[i], a[i+1], p1);
21         if (t1 == 0) {
22             auto minx = min(a[i].first, a[i+1].first);
23             auto maxx = max(a[i].first, a[i+1].first);
24             if (minx <= p1.first && p1.first <= maxx) {
25                 return 1;
26             }
27         }
28     }
29     Point p0 = {1, 2147483647};
30     int cnt = 0;
31     for (int i=0; i<n; i++) {
32         Line l = {p0, p1};
33         cnt += cross(l, Line({a[i], a[i+1]}));
34     }
35     return cnt%2;
36 }
37 int main() {
38     int n;
39     cin >> n;
40     vector<Point> a(n);
41     for (int i=0; i<n; i++) {
42         cin >> a[i].first >> a[i].second;
43     }
44     a.push_back(a[0]);
45     int m = 3;
46     while (m--) {
47         Point p1;
48         cin >> p1.first >> p1.second;
49         cout << go(p1, n, a) << '\n';
50     }
51     return 0;
52 }
53
```

변수이?

결과

메모리

시간

코드 길이

맞았습니다!!

1988 KB

0 ms

1511 B

C++14

```
1 #include <cstdio>
2 #include <vector>
3 #include <algorithm>
4 using namespace std;
5 struct Point {
6     int x, y;
7 };
8 Point a[200000];
9 Point p;
10 int ccw(Point p1, Point p2, Point p3) {
11     long long temp = (long long)(p2.x-p1.x)*(long long)(p3.y-p1.y)-(long long)(p3.x-p1.x)*(long long)(p2.y-
p1.y);
12     if (temp > 0) {
13         return 1;
14     } else if (temp < 0) {
15         return -1;
16     } else {
17         return 0;
18     }
19 }
20 long long dist (Point p1, Point p2) {
21     long long d1 = (long long)(p1.x - p2.x);
22     long long d2 = (long long)(p1.y - p2.y);
23     return d1*d1 + d2*d2;
24 }
25 bool cmp(const Point &u, const Point &v) {
26     int temp = ccw(p, u, v);
27     if (temp == 0) {
28         return dist(p, u) <= dist(p, v);
29     } else {
30         return temp == 1;
31     }
32 }
33 int main() {
34     int n;
35     scanf("%d",&n);
36     for (int i=0; i<n; i++) {
37         scanf("%d %d",&a[i].x,&a[i].y);
38     }
39     p = a[0];
40     for (int i=1; i<n; i++) {
41         if (a[i].y < p.y || (a[i].y == p.y && a[i].x < p.x)) {
42             p = a[i];
43         }
44     }
45     sort(a,a+n,cmp);
46     vector<Point> s;
47     for (int i=0; i<n; i++) {
48         while (s.size() >= 2 && ccw(s[s.size()-2], s[s.size()-1], a[i]) <= 0) {
49             s.pop_back();
50         }
51         s.push_back(a[i]);
52     }
53     printf("%d\n",s.size());
54     return 0;
55 }
```

Handwritten box around lines 25-32 of the code.

Handwritten annotations on the code:

- Line 36: **입력** (Input)
- Line 39: **가장 ↓, ←** (Farthest down, left)
- Line 41: **가장 ↓, ←** (Farthest down, left)
- Line 42: **3개** (3 cases)
- Line 43: **2개** (2 cases)
- Line 44: **반시계** (Counter-clockwise)
- Line 45: **반시계** (Counter-clockwise)
- Line 46: **반시계** (Counter-clockwise)
- Line 47: **반시계** (Counter-clockwise)
- Line 48: **반시계** (Counter-clockwise)
- Line 49: **반시계** (Counter-clockwise)
- Line 50: **반시계** (Counter-clockwise)
- Line 51: **반시계** (Counter-clockwise)
- Line 52: **반시계** (Counter-clockwise)
- Line 53: **반시계** (Counter-clockwise)
- Line 54: **반시계** (Counter-clockwise)
- Line 55: **반시계** (Counter-clockwise)

결과	메모리	시간	코드 길이
맞았습니다!!	2740 KB	48 ms	1282 B

C++14

```
1 #include <iostream>
2 #include <cmath>
3 #include <vector>
4 #include <set>
5 #include <algorithm>
6 using namespace std;
7 struct Point {
8     int x, y;
9     bool operator< (const Point &v) const {
10         if (y == v.y) {
11             return x < v.x;
12         } else {
13             return y < v.y;
14         }
15     }
16 };
17 bool cmp(const Point &u, const Point &v) {
18     return u.x < v.x;
19 }
20 int dist(Point p1, Point p2) {
21     return (p1.x-p2.x)*(p1.x-p2.x) + (p1.y-p2.y)*(p1.y-p2.y);
22 }
23 int main() {
24     ios_base::sync_with_stdio(false);
25     cin.tie(nullptr);
26     int n;
27     cin >> n;
28     vector<Point> a(n);
29     for (int i=0; i<n; i++) {
30         cin >> a[i].x >> a[i].y;
31     }
32     sort(a.begin(), a.end(), cmp);
33     set<Point> candidate = {a[0], a[1]};
34     int ans = dist(a[0], a[1]);
35     int start = 0;
36     for (int i=2; i<n; i++) {
37         Point now = a[i];
38         while (start < i) {
39             auto p = a[start];
40             int x = now.x - p.x;
41             if (x*x > ans) {
42                 candidate.erase(p);
43                 start += 1;
44             } else {
45                 break;
46             }
47         }
48         int d = (int)sqrt((double)ans)+1;
49         auto lower_point = Point({-100000, now.y-d});
50         auto upper_point = Point({100000, now.y+d});
51         auto lower = candidate.lower_bound(lower_point);
52         auto upper = candidate.upper_bound(upper_point);
53         for (auto it = lower; it != upper; it++) {
54             int d = dist(now, *it);
55             if (d < ans) {
56                 ans = d;
57             }
58         }
59         candidate.insert(now);
60     }
61     cout << ans << '\n';
62     return 0;
63 }
64
```

<

y를 기준으로 만든  
인사~

x를 기준으로 (점렬통)

원격거리 구함

↓: 평면 (가장 가까운 두 점  
사이의 거리)의 제곱

Candidate: 마지막 점과  
거리가 가장 가까워지도록  
↓이하

Start 번째부터  
들어간 것

→ 거리가 제일

현재가 들어있는 것을  
7번째로 x좌표로  
차별 ↓ 이하?  
후!

6번 반복

EN

NO  
NO  
NO



## C++14

```

1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4 using namespace std;
5 int main() {
6     ios_base::sync_with_stdio(false);
7     cin.tie(nullptr);
8     int n;
9     cin >> n;
10    vector<pair<int,int>> a(2*n);
11    for (int i=0; i<n; i++) {
12        int x, y;
13        cin >> x >> y;
14        a[i] = make_pair(x, 1);
15        a[i+n] = make_pair(y, -1);
16    }
17    sort(a.begin(), a.end());
18    int ans = 0;
19    int cnt = 0;
20    for (auto &p : a) {
21        cnt += p.second;
22        if (cnt > ans) ans = cnt;
23    }
24    cout << ans << '\n';
25    return 0;
26 }
27

```

Handwritten annotations:
 

- Arrows pointing to the `1` in `make_pair(x, 1)` and the `-1` in `make_pair(y, -1)`.
- A box around `p.second` in the loop.
- A box around the `if` statement.
- A box around `ans` in the `cout` statement.
- Handwritten text `x 3 3 3 3` next to the `sort` line.

결과

메모리

시간

코드 길이

맞았습니다!!

17616 KB

472 ms

547 B

## C++14

```

1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4 using namespace std;
5 int main() {
6     ios_base::sync_with_stdio(false);
7     cin.tie(nullptr);
8     int n;
9     cin >> n;
10    vector<pair<int,int>> a(2*n);
11    for (int i=0; i<n; i++) {
12        int x, y;
13        cin >> x >> y;
14        a[i] = make_pair(x,1);
15        a[i+n] = make_pair(y,-1);
16    }
17    sort(a.begin(),a.end());
18    int ans = 0;
19    int cnt = 0;
20    int last = 0;
21    for (auto &p : a) {
22        if (p.second == 1 && cnt == 0) {
23            last = p.first;
24        }
25        cnt += p.second;
26        if (p.second == -1 && cnt == 0) {
27            ans += p.first - last;
28        }
29    }
30    cout << ans << '\n';
31    return 0;
32 }
33

```

Handwritten annotations:

- Line 17:  $\underline{\hspace{1cm}} \times$
- Line 20:  $\boxed{\text{last}} = 0;$  with arrow pointing to "시작" (start)
- Line 21:  $\text{for (auto \&p : a) \{}$  with arrow pointing to "시작" (start)
- Line 22:  $\text{if (p.second == 1 \&\& cnt == 0) \{}$  with arrow pointing to "시작" (start)
- Line 23:  $\text{last = p.first;}$  with arrow pointing to "시작" (start)
- Line 25:  $\text{cnt += p.second;}$  with arrow pointing to "0 → 1"
- Line 26:  $\text{if (p.second == -1 \&\& cnt == 0) \{}$  with arrow pointing to "1 → 0"
- Line 27:  $\text{ans += p.first - last;}$  with arrow pointing to "2" (result)

결과

메모리

시간

코드 길이

맞았습니다!!

17616 KB

480 ms

697 B

끝

---



# 코드 플러스

<https://code.plus>

- 슬라이드에 포함된 소스 코드를 보려면 "정보 수정 > 백준 온라인 저지 연동"을 통해 연동한 다음, "백준 온라인 저지"에 로그인해야 합니다.
- 강의 내용에 대한 질문은 코드 플러스의 "질문 게시판"에서 할 수 있습니다.
- 문제와 소스 코드는 슬라이드에 첨부된 링크를 통해서 볼 수 있으며, "백준 온라인 저지"에서 서비스됩니다.
- 슬라이드와 동영상 강의는 코드 플러스 사이트를 통해서만 볼 수 있으며, 동영상 강의의 녹화와 다운로드, 배포와 유통은 저작권법에 의해서 금지되어 있습니다.
- 다른 경로로 이 슬라이드나 동영상 강의를 본 경우에는 [codeplus@startlink.io](mailto:codeplus@startlink.io) 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.