Code for Jarvis Algorithm

Run on IDE (https://ide.codingblocks.com/#/s/6518)

```
#include <bits/stdc++.h>
using namespace std;

struct Point
{
    int x, y;
};

int orientation(Point p, Point q, Point r)
{
    int val = (q.y - p.y) * (r.x - q.x) -
              (q.x - p.x) * (r.y - q.y);

    if (val == 0) return 0;
    return (val > 0)? 1: 2;
}

void convexHull(Point points[], int n)
{
    if (n < 3) return;
    vector<Point> hull;
    int l = 0;
    for (int i = 1; i < n; i++)
        if (points[i].x < points[l].x)
            l = i;

    int p = l, q;
    do
    {
        hull.push_back(points[p]);
        q = (p+1)%n;
        for (int i = 0; i < n; i++)
        {
           if (orientation(points[p], points[i], points[q]) == 2)
               q = i;
        }
        p = q;

    } while (p != l);

    for (int i = 0; i < hull.size(); i++)
        cout << "(" << hull[i].x << ", "
              << hull[i].y << ")\n";
}

int main()
```

```
{
    Point points[] = {{0, 3}, {2, 2}, {1, 1}, {2, 1},
                      {3, 0}, {0, 0}, {3, 3}};
    int n = sizeof(points)/sizeof(points[0]);
    convexHull(points, n);
    return 0;
}
```