

Otoczka wypukła

Algorytm Jarvisa i Grahama

Za pomocą wybranego języka programowania wygeneruj chmurę punktów przeznaczonych do utworzenia otoczki wypukłej. Możesz skorzystać z poniższych podpowiedzi (algorytm Jarvisa):

```
void jarvisConvexHull(){

    Polygon curPoly = dm.polygons.get(dm.selectedPolygon);

    dm.polygons.get(dm.selectedPolygon).edges.clear();

    if(curPoly.nodes.size() > 2){

        dm.convexHull = new ArrayList<>(); //clear previously found convex hull

        Node rightPoint = curPoly.nodes.get(0); //point located on the rightmost

        //find point on the right
        for(int i = 0; i < curPoly.nodes.size(); i++){

            if(curPoly.nodes.get(i).x >= rightPoint.x){

                rightPoint = curPoly.nodes.get(i);

            }

        }

        //=====
        dm.convexHull.add(rightPoint); //add first point to convex hull

        Node nextPoint = curPoly.nodes.get(0);

        int i = 0;

        Random rand = new Random(System.currentTimeMillis());

        do{

            nextPoint = curPoly.nodes.get(i % curPoly.nodes.size());

            if(rightPoint.x == curPoly.nodes.get(0).x && rightPoint.y == curPoly.nodes.get(0).y)
                nextPoint = curPoly.nodes.get(1 + rand.nextInt(curPoly.nodes.size()-1));

            for(int j = 0; j < curPoly.nodes.size(); j++){

                if(onLeft(curPoly.nodes.get(j), nextPoint, dm.convexHull.get(i)) == 2){ //check if there's
                    any point on the left of the line
                    nextPoint = curPoly.nodes.get(j);

                }

            }

        }

    }

}
```

```

    }

    dm.convexHull.add(nextPoint);
    i++;

    curPoly.edges.add(new Edge(dm.convexHull.get(i-1), dm.convexHull.get(i)));

}while((nextPoint.x != dm.convexHull.get(0).x && nextPoint.y != dm.convexHull.get(0).y));

//    for(int k = 0; k < dm.convexHull.size()-2; k++){
//
//        //dm.convexHull.get(k).connections.add(dm.convexHull.get(k+1));
//        curPoly.nodes.get(k).connections.add(dm.convexHull.get(k+1));
//    }

//    curPoly.nodes.clear();
//    curPoly.nodes.addAll(dm.convexHull);

}

}

//metoda onLeft
int onLeft(Node p1, Node p2, Node p3){

    //p1 is investigated point
    //p2-p3 is line

    int isLeft = 0;

    double eq = (p3.x - p2.x) * (p1.y - p2.y) - (p1.x - p2.x) * (p3.y - p2.y);
    //System.out.println("Eq: " + eq);

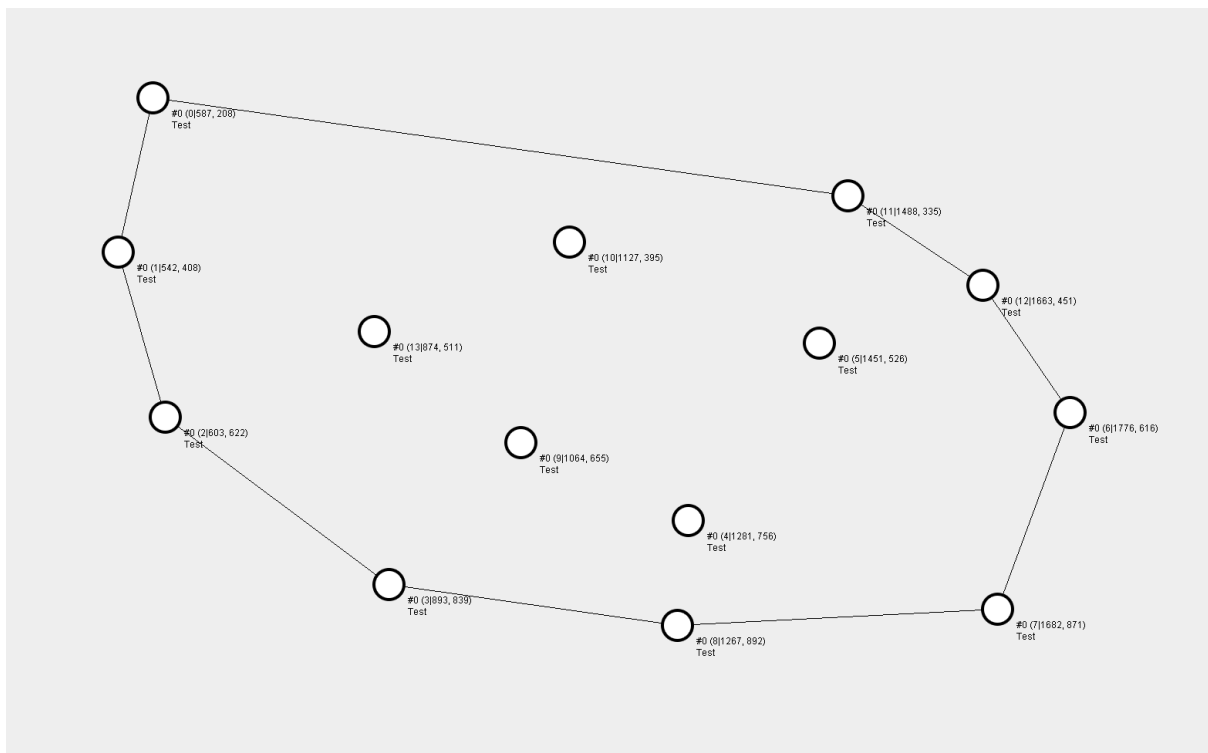
    if(eq > 0)
        isLeft = 2;

    if(eq < 0)
        isLeft = 1;

    return isLeft;

}

```



Przykład działania programu.

Zaimplementuj samodzielnie dowolną inną metodę wyznaczania otoczki wypukłej.