In, On, or Out

We consider a point x and a convex polygon U in the plane. The vertices are assumed to be given clockwise. Furthermore, no three vertices are collinear.

The purpose of this note is to give a smooth development of an algorithm that determines whether the point x is in the interior, on the boundary, or in the exterior of the polygon U. The algorithm should, moreover, be such that the number of computational steps required is proportional to the logarithm of the number of vertices of the polygon.

+

For a (simple) polygon Z we use the notations in.Z , on.Z , and out.Z as shorthands for the conditions x is in the interior of Z , x is on the boundary of Z , and x is in the exterior of Z respectively. Furthermore, we assume the availability of a three-valued function p.e yielding the clockwise position of the point x with respect to a directed line e :

p.e = 1 if x is on the clockwise side of e ,
p.e = 0 if x is on e , and
p.e = −1 if x is on the counterclockwise side of e .

To begin with, we give an algorithm that is linear in the number of vertices of U. With e in the domain of the (clockwise-) directed boundary lines of U, we have on account of U's convexity

$$\begin{aligned} in.U &\equiv (\underline{A}e:: p.e = 1) \\ on.U &\equiv (\underline{A}e:: p.e \geq 0) \wedge (\underline{E}e:: p.e = 0) \\ out.U &\equiv (\underline{E}e:: p.e = -1) \end{aligned}$$

The position of x with respect to U is now computed by computing the minimum of the p-values: in.U, on.U, and out.U then correspond to this minimum being equal to 1, 0, and -1 respectively.

+

For a logarithmic solution we propose to bisect the polygon until a triangle is left. The question then is how to bisect precisely and under control of which invariant.

Let V be a convex polygon (no three vertices collinear and the vertices being given clockwise, but we will not repeat that all the time), with at least four vertices. For y and z two nonadjacent vertices of V we conclude on account of V's convexity that the interior of the line segment yz is in the interior of V. The line connecting y and z divides V into two convex polygons V0 and V1. Now we can state

in.V ≡ in.V0 ∨ in.V1 ∨ x is in the interior of the line segment yz

$$\text{on}.V \equiv (\text{on}.V0 \vee \text{on}.V1) \wedge \neg(x \text{ is in the interior of the line segment } yz)$$

$$\text{out}.V \equiv \text{out}.V0 \wedge \text{out}.V1 \qquad (0)$$

From these three relations we conclude that the notion out is by far the simplest and is likely to give rise to the fewest case analyses in the bisection procedure. Therefore we choose as an invariant relation $P0 \wedge P1$, given by

$P0$: V is a convex polygon with at least three vertices. and all vertices are vertices from U.

$P1$: $\text{out}.U \equiv \text{out}.V$.

The corresponding program becomes

```
   V := U
 ; do the number of vertices of V ≠ 3
   →    select y and z
      ; if B0 → V := V0
        ▯ B1 → V := V1
        fi
   od
 ; triangle analysis .
```

The invariance of $P0$ follows from the way in which $V0$ and $V1$ are constructed from V, y, and z. The invariance of $P1$ has to follow from a proper choice of $B0$ and $B1$. We compute $B0$. In order that the assignment $V := V0$ establishes $P1$ the precondition should imply

$$\text{out}.U \equiv \text{out}.V0 \; .$$

A precondition is (see P1 and (0))

$$out.U \equiv out.V0 \wedge out.V1$$

so that for B0 the choice

$$out.V0 \equiv out.V0 \wedge out.V1$$

or, equivalently,

$$out.V0 \Rightarrow out.V1$$

will do. With the assumption that the line segment from y to z is a clockwise edge of V0 this condition is implied by – for a proof, see the appendix –

$$p.(\text{line from } y \text{ to } z) \geq 0$$

By symmetry, we take for B1

$$p.(\text{line from } z \text{ to } y) \geq 0 \quad .$$

Next we give the triangle analysis. With the linear algorithm we compute in.V, on.V, and out.V for the triangle V. If out.V holds, then, on account of P1, also out.U holds. If in.V holds, then, on account of P0 and U's convexity, also in.U holds. If on.V holds (i.e. each edge of the triangle has a p-value ≥ 0, and one or two p-values $= 0$) we distinguish two cases:

- two edges have a p-value $= 0$; x then coincides with the vertex joining these two edges, hence it coincides (see P1) with a vertex of U, hence on.U holds
- one edge has a p-value $= 0$; x then is in the interior of that edge. If that

edge is an edge of U, on.U holds. If that edge is not an edge of U, it is an edge (see P0) between two nonadjacent vertices of U, so that (U is convex) in.U holds.

+

This completes the derivation of the algorithm. As was to be expected, some case analysis could not be avoided: it is more or less baked in in the problem statement. The rule we followed then was to try to isolate the case analysis in the simplest entourage encountered during the development, here: the triangle after the repetition. Had we permitted ourselves case analysis inside the repetition we would have had to deal with the arbitrary convex polygon V, which is more complicated than a triangle. The main reason, however, for having written this note is to indicate that we have discovered how to minimize case analysis, viz. by doing justice to the simplicity of the shape of formula (0).

W.H.J. Feijen

Austin,
27 May 1985

Appendix

First we observe that the line from y to z is a clockwise boundary line of $V0$, the line from z to y is a clockwise boundary line of $V1$, and p.(line from y to z) = − p.(line from z to y).

Secondly we consider two cases for p.(line from y to z) ≥ 0 :

- p. (line from y to z) = 1, hence p. (line from z to y) = −1, hence (because line from z to y is a clockwise boundary line of V1) out. V1, hence out. V0 $\Rightarrow$ out. V1
- p. (line from y to z) = 0. Again we distinguish two cases
  - x on the line segment yz (bounds included), hence on. V0 and on. V1, hence $\neg$ out. V0 and $\neg$ out. V1, hence out. V0 $\Rightarrow$ out. V1
  - x not on the line segment yz, hence (because line between y and z is a boundary line of V1) out. V1, hence out. V0 $\Rightarrow$ out. V1.

I am not particularly pleased with this proof, and I would rather have a much more calculational argument. But the development of an adequate such a calculus falls outside the scope of this note

(End of Appendix.)

(End of In, On, or Out.)