Ray shooting from an edge. Given a n-vertex simple polygon P and an edge e of P, show how to construct a data structure to answer the following query in $O(\log n)$ time and $O(n)$ space: Given a ray r whose origin lies on e and which is directed into the interior or P, return the edge of P that r hits first.

PROOF. (This is only a sketch of the proof. High level idea is that we will convert a query into a point location query in the dual space.)

For simplicity, assume that the fixed edge e is oriented vertically. Given any point p, let p^* denote the dual line of p. For any ray r emanating from e, let $l(r)$ denote the line containing r, the dual of which is denoted by $l^*(r)$ (which is a point). The set of lines (rays) passing through edge $e = \langle p, q \rangle$ are dualized into the set of points in the slab between two horizontal lines $p*$ and q^*.

Now, take an arbitrary edge $f \neq e$. Let $R(f)$ be the set of all rays which emanates from e and which hits f before it hits any other edge in P. Obviously, $l(r)$ is above a subset of points from P, call it P_1, and below the remaining set of points $P_2 = P - P_1$. In other word, point $l^*(r)$ lies above all dual lines from P_1^* and below all dual lines from P_2^*. It is therefore easy to see that the dual of $R(f)$ is a convex region which lies within the slab between p^* and q^*. The dual of $R(f)$'s for all edges from P form a convex subdivision of the slab. Since there are n convex regions (each corresponding to some edge from P), the overall complexity of the subdivision in the dual space is $O(n)$ (why?). To find out which edge in P is hit first by a query ray r emanating from e is the same as locating the point $l^*(r)$ in this subdivision, which can be done in $O(\log n)$ time using a linear size data structure. □