Laptops

Problem statement: Laptops (link)
Implementation: Code (link)

Our approach is to sort the laptops by price, then do a linear search. After sorting (increasingly), let $L_i.quality$ denote the quality of the i-th laptop. Also, let max_i denote the maximum quality up to and including the i-th laptop. Then we can visit each laptop, from cheapest to most expensive. On each visit, we check whether $L_i.quality < max_{i-1}$ and update max_i. If at any point the the inequality $L_i.quality < max_{i-1}$ is true, we can output "Happy Alex" and terminate. Otherwise, at the end of this proccess, we output "Poor Alex".

Runtime complexity: $O(nlog(n))$, the sort subroutine dominates.
Memory complexity: $O(n)$