21-1 Off-line minimum

The off-line minimum problem asks us to maintain a dynamic set $T$ of elements from the domain $\{1, 2, \ldots, n\}$ under the operations INSERT and EXTRACT-MIN. We are given a sequence $S$ of $n$ INSERT and $m$ EXTRACT-MIN calls, where each key in $\{1, 2, \ldots, n\}$ is inserted exactly once. We wish to determine which key is returned by each EXTRACT-MIN call. Specifically, we wish to fill in an array $\text{extracted}[1 \ldots m]$, where for $i = 1, 2, \ldots, m$, $\text{extracted}[i]$ is the key returned by the $i$th EXTRACT-MIN call. The problem is “off-line” in the sense that we are allowed to process the entire sequence $S$ before determining any of the returned keys.

a. In the following instance of the off-line minimum problem, each operation INSERT$(i)$ is represented by the value of $i$ and each EXTRACT-MIN is represented by the letter $E$:

$$4, 8, E, 3, E, 9, 2, 6, E, E, E, 1, 7, E, 5.$$ 

Fill in the correct values in the $\text{extracted}$ array.

To develop an algorithm for this problem, we break the sequence $S$ into homogeneous subsequences. That is, we represent $S$ by

$$I_1, E, I_2, E, I_3, \ldots, I_m, E, I_{m+1},$$

where each $E$ represents a single EXTRACT-MIN call and each $I_j$ represents a (possibly empty) sequence of INSERT calls. For each subsequence $I_j$, we initially place the keys inserted by these operations into a set $K_j$, which is empty if $I_j$ is empty. We then do the following: