Algorithm 2 ikNNQ

1: **function** IKNNQ(query point q, k, indoor index $\mathcal{T}$)
2:     result set R; candidate object set C;
3:     $(R_1^o, R_1^p) \leftarrow kSeedsSelection(q, k)$;   // **Phase 1: filtering**
4:     kbound $\leftarrow max_{O \in R_1^o}\{|q, O|_I.TLU\}$; // (Lemma 3)
5:     $(R_2^o, R_2^p) \leftarrow RangeSearch(q, kbound, \mathcal{T})$;
6:     Dijkstra(R_2^p);   // **Phase 2: subgraph**
7:     **for** each object O in R_2^o **do**   // **Phase 3: pruning**
8:         $[O.l, O.u] \leftarrow [|q, O|_{minI}, |q, O|_{maxI}]$;   // (Table III)
9:     Find object O_k which has the k-th shortest $O.u$; set $C = \emptyset$;
10:     **for** each $O \in R_2^o$ **do**
11:         **if** $O.u < O_k.l$ **then** $R = R \cup \{O\}$
12:         **else**
13:             **if** $O.l \leq O_k.u$ **then** $C = C \cup \{O\}$
14:     **for** each $O \in C$ **do**   // **Phase 4: refinement**
15:         Calculate $|q, O|_I$;
16:     Sort objects in C by $|q, O|_I$ in ascending order and add top $k - |R|$ objects to R;
17:     **return** R.