function SEMI-NEIGHBORHOOD-JOIN (indoor index $\mathcal{T}_{\mathcal{Q}}$, indoor index $\mathcal{T}_{\mathcal{O}}$, parameter k)
result set R; candidate object set C; $\triangleright$ Global variables
for each of T_Q's leaf nodes $\mathcal{Q}$ **do** $\triangleright$ Phase 1: filtering
if $\mathcal{Q}._{count}$ equals to 0 **then**
Add $\mathcal{Q}$ to C;
for each partition Q^P in C **do**
$(R_1^o, R_1^p) \leftarrow$kSeedsSelection$(Q^P, k)$;
kbound $\leftarrow max_{O \in R_1^o}\{|Q^P, O|_I.TLU\}$; $\triangleright$ Lemma 5
$(R_2^o, R_2^p) \leftarrow$RangeSearch($Q^P$, kbound + $Q^P._{r_{max}}$,T_O);
Dijkstra(R_2^p); $\triangleright$ Phase 2: subgraph
for each object Q in partition Q^P **do**
$C_O \leftarrow \emptyset$ $\triangleright C_O$ is a set for candidate objects in $\mathbb{O}$
$R_O \leftarrow \emptyset$ $\triangleright R_O = kNN(Q)$
for each object O in R_2^o **do** $\triangleright$Phase 3: pruning
$[O.l, O.u] \leftarrow [|Q, O|_{minI}, |Q, O|_{maxI}]$; $\triangleright$ Table 2
Find object O_k which has the k-th shortest $O.u$;
for each $O \in R_2^o$ **do**
if $O.u < O_k.l$ **then** $R_O = R_O \cup \{O\}$
else
if $O.l \leq O_k.u$ **then** $C_O = C_O \cup \{O\}$
for each $O \in C_O$ **do** $\triangleright$Phase 4: refinement
Calculate $|Q, O|_I$;
Sort objects in C_O by $|Q, O|_I$ in ascending order and add top $k - |R_O|$ objects to R_O;
$R \leftarrow R \cup \{Q, O\}_{O \in R_O}$
return R;