Selection Algorithm

Assume all elements distinct.

procedure Select(S, k):
{ Input: S a set of keys, $|S| = n$; $1 \leq k \leq n$, an integer }
{ Output: $a \in S$, with k keys in $S \leq a$ and $n - k$ keys $> k$ }

$S^- = \emptyset$ { S^- contains those found $<$ splitter }
$S^+ = \emptyset$ { S^+ contains those found $>$ splitter }
choose $a_i \in_R S$
for each $a_j \in S$, $j \neq i$ **do**
  if $a_j < a_i$
    then put a_j in S^-
    else put a_j in S^+
if $|S^-| = k - 1$
  then return(a_i)
  else if $|S^-| \geq k$
    then Select(S^-, k)
    else Select$(S^+, k - 1 - |S^-|)$

Quicksort — Same form as Select

Assume all elements distinct.

procedure Quicksort(S):
{ Input: S a set of keys, $|S| = n$ }
{ Output: The keys from S in nondecreasing order }
    if $|S| \leq 3$
        then Sort S and output the sorted list
        else
            $S^- = \emptyset$; $S^+ = \emptyset$
            choose $a_i \in_R S$
            for each $a_j \in S$, $j \neq i$ **do**
                if $a_j < a_i$
                    then put a_j in S^-
                    else put a_j in S^+
            Output Quicksort(S^-)
            Output a_i
            Output Quicksort(S^+)

Quicksort — Same form as Select

Assume all elements distinct.

procedure Modified Quicksort(S):
    if $|S| \leq 3$
        then Sort S and output the sorted list
        else
            while no central splitter has been found
                $S^- = \emptyset$; $S^+ = \emptyset$
                choose $a_i \in_R S$
                for each $a_j \in S$, $j \neq i$ **do**
                    if $a_j < a_i$
                    then put a_j in S^-
                    else put a_j in S^+
                if $|S^-| \geq |S|/4$ and $|S^+| \geq |S|/4$
                    then a_i is a central splitter
            Output Modified Quicksort(S^-)
            Output a_i
            Output Modified Quicksort(S^+)