QUICK SORT

Quick Sort (A, BEG, END):
Desciption: Here **A** is an unsorted array having **N** elements. **BEG** is the lower bound and **END** is the upper bound.

1. If (BEG < END) then
2. Find the element that divides the array into two parts using subfunction **Partition( )**.
3. Quick Sort (Left Half)
4. Quick Sort (Right Half)
[End of If]
5. Exit

Partition ( ):

1. Set LEFT = BEG, RIGHT = END and LOC = BEG
2. Beginning with the element pointed by RIGHT, scan the array from right to left, comparing each element with the element pointed by LOC until:
(a) Element smaller than the element pointed by LOC is found.
(b) Interchange elements pointed by LOC and RIGHT.
(c) If RIGHT becomes equal to LOC, terminate the subfunction Partition ( ).
3. Beginning with the element pointed by LEFT, scan the array from left to right, comparing each element with the element pointed by LOC until:
(a) Element greater than the element pointed by LOC is found.
(b) Interchange elements pointed by LOC and LEFT.
(c) If LEFT becomes equal to LOC, terminate the subfunction Partition ( ).
4. Exit