

SELECTION-SORT( $A, n$ )

1. **for**  $i = 1$  **to**  $n - 1$
2.      $smallestIndex = i$
3.     **for**  $k = i$  **to**  $n$
4.         **if**  $A[k] < A[smallestIndex]$
5.              $smallestIndex = k$
6.      $temp = A[smallestIndex]$
7.      $A[smallestIndex] = A[i]$
8.      $A[i] = temp$

The loop-invariant is that after the  $i^{th}$  iteration, all the elements  $A[1 : i]$  will be in sorted order and smaller than all the elements in  $A[i + 1 : n]$ .

Here, the worst case running time and the best case running time are exactly the same,  $\Theta(n^2)$ , because regardless of what order the array is in the algorithm performs a total  $n(n - 1)/2$  comparisons after having looked for the smallest element in each sub-array.