Selection sort. There are many algorithms for sorting. Selections sort is simple but performs poorly. The idea behind selection sort is simply to repeatedly exchange the largest element in the list to be sorted with the last element in the list. Diminish the size of list by one each iteration. Thus, in the first iteration, the largest element is placed at the end of the list. In the second iteration, the next largest is placed next to the last position in the list. The process of selecting the largest element in an ever-diminishing list continues until the list is sorted in ascending order.

Example: Shaded areas in the list below mark the portion of the list that is sorted.

-15	29	-54	16	32	105	-12	6
-15	29	-54	16	32	6	-12	105
-15	29	-54	16	-12	6	32	105
-15	6	-54	16	-12	29	32	105
-15	6	-54	-12	16	29	32	105
-15	-12	-54	6	16	29	32	105
-15	-54	-12	6	16	29	32	105
-54	-15	-12	6	16	29	32	105
-54	-15	-12	6	16	29	32	105

Algorithm:

1. Create variable *eol*, end of list. Assign the index of the last element in the list to variable *eol*.
2. Test: Is the value of *eol* greater than or equal to the second element in the list? Yes, continue to step 2.1. No, go to step 3.
   - 2.1. Create variable *iom*, index of maximum. Assign index of the first element in the array to *iom*.
   - 2.2. Create variable *i* used to traverse the list. Start *i* at the second element in the array.
   - 2.3. Test: Is the value of *i* less than or equal to the value of *eol*. Yes, continue to step 2.3.1. No, go to step 2.4.
     - 2.3.1. Test: Is the value of the current element, the i^{th} element greater than the element whose index is *iom*. Is the current element greater than the maximum? The index of the maximum is variable *iom*. Yes, assign the index of the current element, *i*, to variable *iom*, making the current element the maximum.
     - 2.3.2. Increment the value of variable *i*.
     - 2.3.3. Go to step 2.3
   - 2.4. Exchange the elements whose index positions are given by variable *iom* and *eol*.
   - 2.5. Decrement the value of variable *eol*.
   - 2.6. Go to step 2
3. All done. The list is sorted.

Example: The list is implement in array *L* having a range of indexes from 0 to 7.

L[0]	***L[1]***	***L[2]***	***L[3]***	***L[4]***	***L[5]***	***L[6]***	***L[7]***	***iom***	***eol***
-15	29	-54	16	32	105	-12	6	5	7
-15	29	-54	16	32	6	-12	105	4	6
-15	29	-54	16	-12	6	32	105	1	5
-15	6	-54	16	-12	29	32	105	3	4
-15	6	-54	-12	16	29	32	105	1	3
-15	-12	-54	6	16	29	32	105	1	2
-15	-54	-12	6	16	29	32	105	0	1
-54	-15	-12	6	16	29	32	105		
-54	-15	-12	6	16	29	32	105		