

**Armstrong Atlantic State University**  
**Engineering Studies**  
**MATLAB Marina – Sorting Exercises**

1. Modify the `selectionSort` function of Figure 1 to sort the data in decreasing order (highest to lowest). Test the modified function for the array [5, 4, 10, 2, 4, 6, 3, 7, 8, 1].

```
function result = selectionSort(data)

result = data;
% search list number times equal to its size
for pass = 1 : 1 : length(result)-1
    smallindex = pass;
    % search remaining unordered sublist for next smallest element
    for k = pass + 1 : 1 : length(result)
        if (result(k) < result(smallindex))
            % update next smallest element if necessary
            smallindex = k;
        end
    end
    % put next smallest element in correct place by swapping
    if (smallindex ~= pass)
        temp = result(pass);
        result(pass) = result(smallindex);
        result(smallindex) = temp;
    end
end
end
```

Figure 1, selectionSort Function

2. Modify the `selectionSort` function of Figure 1 to count the number of comparisons and the number of swaps performed during the sorting. For test purposes only, display the number of comparisons and number of swaps performed at the end of the function. Test the modified function for the arrays  $x = [5, 4, 10, 2, 9, 6, 3, 7, 8, 1]$  and  $x = \text{round}(100 \cdot \text{rand}(1,100))$ .
3. Use MATLAB's `sort` function to sort the array [5, 4, 10, 2, 9, 6, 3, 7, 8, 1] in ascending order.
4. Use MATLAB's `sort` function to sort the array [5, 4, 10, 2, 9, 6, 3, 7, 8, 1] in descending order.
5. Given the array of numbers [19, 41, 23, 8, 71, 20, 65, 34], show the arrays produced during the operation of a selection sort, a merge sort, and a quick sort to sort the array in increasing order. For the quick sort, assume that the first element of the arrays is used as the pivot.

6. Write a MATLAB function `selectionSortStrings` that will sort a 1D cell array of strings. The cell array is assumed to be depth one. Hints: for comparisons remember to extract the contents of the cell containers from the cell array and use the MATLAB `strcmp` function rather than a logic equal comparison for any string comparisons.

Last modified Thursday, November 07, 2013



This work by Thomas Murphy is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License](https://creativecommons.org/licenses/by-nc-nd/3.0/).