Prof. Pietro S. Oliveto

CS217 Data Structures and Algorithm Analysis
Fall term 2024

Exercise Sheet 1

Handout: September 10th — Deadline: 4pm September 17th

SelectionSort sorts by repeatedly finding the smallest element amongst those not yet sorted, and swaps it with the first number in the unsorted part of the array.

```
Selection-Sort($A$)
1: $n = A$.length
2: for $j = 1$ to $n - 1$ do
3:     smallest $= j$
4:     for $i = j + 1$ to $n$ do
5:         if $A[i] < A$[smallest] then smallest $= i$
6:     exchange $A[j]$ with $A$[smallest]
```

Question 1.1 (0.25 marks)

To get a feeling of how SelectionSort works, assume it is run on the array

24	5	6	23	42	45	2	1	8

Write down the contents of the array after every iteration of the for loop in line 2.

Question 1.2 (0.5 marks)

Show the correctness of SelectionSort when run on an array of n different elements (any array, not just the instance from Question 1.1). Find a loop invariant for the loop in line 2 that implies that at termination the array is sorted. Show that this invariant holds at initialisation, and that if it is true before an iteration of the loop, it remains true before the next iteration. Show that the loop invariant at termination implies that the array is sorted.

Question 1.3 (0.5 marks)

Assume for simplicity that one execution of each line of the algorithm takes time 1 (that is, in the notation for analysing InsertionSort, $c_1 = c_2 = \cdots = 1$). Give the best-case and the worst-case running time of SelectionSort. How does this compare to best-case and worst-case times of InsertionSort.

Hint: Notice that in Line 5 the **if** part of the statement may be executed but not the **then** part. The question asks you to consider that the execution of each line takes time 1, so you can ignore the fact that the **then** may not be executed in some iteration of the loop i.e., line 5 costs 1 each time independent of whether the **then** is executed. However, if you wish to split Line 5 into two lines (i.e. Line 5.1 and Line 5.2) each of cost 1 you can also do so.

Programming Question 1.4 (0.25 marks)

Implement InsertionSort and SelectionSort and do the "Double Median", and "Lucky Number" programming exercises on Judge.