Algorithms and Computational Thinking
Autumn 2016

Thursday, 20th October 2016

Exercise 5 - Algorithms and Computational Complexity

1. A wants to write a program to find the minimum number in a list. She writes the following piece of code.

```
x=[5,6,5,9,3,8,1,2] #Declare a list containing some elements
z=len(x)-1 #len(x) returns the size of a list
i=0
for i in range(z) :
    if x[i]<x[i+1] :
        y=x[i]
    else :
        y=x[i+1]
print(y)
```

Execute the above code and check the result. You will notice that the code is able to find the minimum number in the list. However, B wants to test the code and changes the list elements to x=[2,3,5,9,1,0,2,3], and the above program fails to find the minimum number. Your tasks are the following : **(You are free to choose any language of your choise)**

1. What is the bug in the above program ?. Change the above program such that it is able to find the minimum number in the list entered by B maintaining a linear complexity O(n).
2. Add functionality in the above program to find the maximum and also the average number of the list. Write three different functions calculating minimum, maximum and average. In this case what is the complexity involved ? How can the complexity be reduced ?

2. Implement a bubble sort algorithm to sort the lista = [12, 5, 13, 8, 9, 65]. A pseudocode is presented below , refer it and translate it in to a working program in any language of your choice.

```
Bubble Sort
for all elements of list
    if list[i] > list[i+1]
        swap(list[i], list[i+1])
    end if
end for
```