COMP 4030/6030 -- Assignment 7

**Due: Tuesday 11/7/2017**

**This is a coding assignment. Mail your work to the TA:**

* The name your solution file should be the same as your UID, plus a .py extension. For example, if your UID is jsmith (i.e. your email is [jsmith@memphis.edu)](mailto:jsmith@memphis.edu)), then your solution file should be **jsmith.py**.
* In the file, put your full name, COMP 4030 or COMP 6030, and Assignment 7.
* Send your solution to the TA (Quang Tran, [qmtran@memphis.edu)](mailto:qmtran@memphis.edu)) with the subject line “**COMP 4030 Assignment 7**”.

1. (40 Points) Write a Python function that takes as input a list **A** and an index **j** and returns the maximum sum of consecutive numbers in the list that ends at index **j**.

For example, if A = [-20,50,-30,40,10,5,-50], then

maxsum\_ending\_at\_j(A, 0) returns -20

maxsum\_ending\_at\_j(A, 1) returns 50

maxsum\_ending\_at\_j(A, 2) returns 20

maxsum\_ending\_at\_j(A, 3) returns 60

maxsum\_ending\_at\_j(A, 4) returns 70

maxsum\_ending\_at\_j(A, 5) returns 75

maxsum\_ending\_at\_j(A, 6) returns 25

Implement maxsum\_ending\_at\_j(A, j) based on the following hints/questions:

* If j is equal to 0, then what should the function return?
* The max sum that ends at j can either (a) starts at j or (b) starts at an index less than j.
* If the max sum starts at index j, then the return value is L[ j ]. Right?
* If the max sum that starts at index less than j, then the return value is what?
* You can correctly determine which case it is (a) or (b) by looking at the maxsum ending at j-1.
* Look at the example above and see if you can determine max sum ending at j based on the max sum ending at j-1.
* Your function should be recursive.

1. (40 Points) Write the running time equation for your implementation of maxsum\_ending\_at\_j in question 1.
2. (20 Points) Write a Python function that takes as input a list A and returns the maxisum sum of consecutive numbers in the list. Your function should call maxsum\_ending\_at\_j. In this problem, you can assume that maxsum\_ending\_at\_j is implemented correctly.

As an example, maxsum([-20,50,-30,40,10,5,-50]) returns 75, because 50-30+40+10+5 is 75.

1. (10 Points) Write the running time equation for your implementation of maxsum in question 3.
2. (10 Points) Your implementation of maxsum in question 3 can run faster if you storie values that are computed more than once. Modify your implementation of maxsum\_ending\_at\_j (question 1) so that maxsum (question 3) runs in linear time, (n).