Week 5 tute solutions:

Question 1)

What is the computational complexity of this algorithm? Attempt to prove it formally!

$O(2^n)$

$T_1 = 1$

$T_2 = 1$

$T_N = T_{n-1} + T_{n-2}$

$T_N < 2 * T_{n-1}$, given that $T_{N-1} > T_{N-2}$

$T_N < 2^2 * T_{n-2}$

$T_N < 2^3 *$ Tn-3

…

$T_N < 2^N * 1$

Hence the complexity of this algorithm is bounded by $O(2^n)$

Can you write a more efficient version that is NOT iterative, but instead single-recursive (rather than double-recursive as in the version above)?

```
def fib(n):

        return _fib(n, 0, 1)

def _fib(n, a, b):

        if n ==0:

                return b

        return _fib(n-1, b, a+b)
```

What is the time complexity of such a single-recursive implementation?
O(n)

Question 2)

Given the following algorithm:
for i from Lo1 to Hi1

do

for j from Lo2 to Hi2

do

body()

end_for

end_for

How many times is body() executed for the following values?
Lo1=1, Hi1=10, Lo2=i, Hi2=10 --- 55

Lo1=0, Hi1= 9, Lo2=0, Hi2=9 --- 100

Lo1=1, Hi1=n, Lo2=i-2, Hi2=i+2 --- 5n

Lo1=0, Hi1=n, Lo2=i, Hi2=2*i --- See below.

T0 = 1

T1 = T0 + 2 = 3

T2 = T1 + 3 = 6

T3 = T2 + 4 = 10

T3 = T1 + 3 + 4

T3 = T0 + 2 + 3 + 4

T3 = 1 + 2 + 3 + 4

Just a sum of natural numbers to N + 1, so

T_N =((N + 1) (N + 1 + 1)) / 2

T_N =(n2 + 3n + 2) / 2

Question 3)

Given an integer n, body() will be run n times where n > 0, and base() will always run once.

function r(n), given an integer of n > 0 will make a single call to body() before recursively calling itself with the value of n-1. As our n value shrinks by 1 every time our function is recursed, n will be greater than zero on n occasions, calling body() on n occasions. No matter what value n our function r is given (even if a negative number), the "else" condition will yield true on exactly one occasion.

Thus,

r(10), body() = 10, base() = 1

r(5), body() = 5, base() = 1

r(1), body() = 1, base() = 1

Question 5)

Longest increasing subsequence problem:

Refer to https://www.youtube.com/watch?v=Ns4LCeeOFS4

Question 6)

Refer to "*Supplementary Notes to Week 3 Lecture: part 1*" under week 3 tab on Moodle or follow the below link.

http://moodle.vle.monash.edu/pluginfile.php/5673656/mod_resource/content/7/lecture03_1.pdf