For the following pairs of functions, decide whether $f = O(g)$, or $f = \Omega(g)$, or $f = \Theta(g)$, and briefly explains.

(a) $f(n) = n + 100, g(n) = 2n + 30$

(b) $f(n) = \sqrt{n}, g(n) = n^{\frac{2}{3}}$

(c) $f(n) = 10 \log_2 n, g(n) = \log_{10}(n^3)$

(d) $f(n) = n^{12}, g(n) = 1.2^n$

(e) $f(n) = 3^n, g(n) = 3.1^n$
2 Pseudocode, and running time analysis. When perform running time analysis, first express the number of computer steps, \( T(n) \), in terms of \( n \) (input size), and then use big-O notations (the three of them) to simplify the expression.

(a) analyze the running time of the following pseudocode

```plaintext
Largest (a[1...n])
if n==1 return a[1]

mid = (1+n)/2
l1 = Largest(a[1...mid])
l2 = Largest (a[mid+1...n])

if (l1>l2)
    return l1
else
    return l2
```

(b) Analyze the running time of the following procedure which calculates the largest one day drop or increase in the price of a certain commodity/stock, given by a list of numbers.

```plaintext
/* a[1] is the price on first day, a[2] is the price on day 2, ...
   return the largest increase/drop (from one day to the next) in the data */

LargestOneDayDifference (a[1...n])

largestChange = 0 //initialization

for i=1 to n-1
    if (abs(a[i+1]-a[i])) > largestChange
        largestChange = abs (a[i+1]-a[i])

return largestChange
```
(c) Analyze the running time of the following pseudocode which has nested loops:

```python
some_func(data[1...n])
    create an array/list of len n-4, sum
    for i=1 to n-4
        sum[i] = 0
        for j=i to i+4
            sum[i] = sum[i] + data[j]
    largest = sum[i]
    for i=1 to n-4
        if (sum[i]>largest)
            largest = sum[i]
```

3 Solve the following recurrence relations using either the Master Theorem or the method of iteration.

(a) \( T(n) = 2T(n/3) + 1 \)

(b) \( T(n) = 5T(n/4) + n \)

(c) \( T(n) = T(n-1) + n, \)
Use the following formula
\[ 1 + c + c^2 + c^3 + ... + c^n = \frac{c^{n+1} - 1}{c - 1}. \]

to show that, if $c$ is a positive real number, then $g(n) = 1 + c + c^2 + ... + c^n$ is:

(a) $\Theta(1)$ if $c < 1$.
(b) $\Theta(n)$ if $c = 1$.
(c) $\Theta(c^n)$ if $c > 1$. 
How many lines does the following program print? Answer the question with a function of $n$ (in $\Theta(.)$ form).

```python
function f(n)
    if n>1:
        printline( "in here");
        printline( "in here");
        f(n/4)
        f(n/4)
```

5 How many lines does the following program print? Answer the question with a function of $n$ (in $\Theta(.)$ form).

```python
function f(n)
    if n>1:
        printline( "in here");
        printline( "in here");
        f(n/4)
        f(n/4)
```