Q1) What is the runtime of the following function? Express your answer using the big-O notation. Show all working [5 Marks]

```
void mystery (n)
{
    If (n > 1)
    {
        print "hello"
        mystery(2n/3)
        for (i:1 to n)
            print "world"
        mystery(n/5)
    }
}
```

Q2) Prove that T(n) is $\Theta(n^4)$ by finding appropriate constants. [5 Marks]

$$(\ \ ) = 1/6\, n^4 - 4n^2$$

Q3) Suppose we have a sorting algorithm with following pseudocode: [10+5 = 15 Marks]

```
If: the sequence length is at most 4,
    Then sort it using bubble sort.
Else:
i. Divide the list into 5 pieces evenly, by scanning the entire list.
ii. (recursively) sort the first 3/5 of the list.
iii. (recursively) sort the last 3/5 of the list.
iv. (recursively) sort the first 3/5 of the list.
```

For example, on the input sequence 1, 5, 3, 2, 4 The first recursive sort produces 1, 3, 5, 2, 4, the second sort produces 1, 3, 2, 4, 5, and the last produces 1, 2, 3, 4, 5.

(a) Write down a runtime recurrence for this sorting algorithm and analyze its asymptotic running time.

(b) Give an example sequence on 5 or 10 integers where this sorting algorithm does not terminate with the correct answer.

Q4) Use a recursion tree to determine a good asymptotic upper bound on following recurrences.

Please see the Appendix of your textbook for using harmonic and geometric series.

a) $T(n) = 2T(n/4) + \sqrt{n}$

b) $T(n) = T(n - 2) + n^2$

c) $T(n) = 10T(n/2) + \Theta(1)$

d) $T(n) = 2T(n/2) + n/\lg n$

e) $T(n) = 2T(n-1) + \Theta(1)$

f) $T(n) = T(n-1) + 1/n$

g) $T(n) = 4T(n/2) + n^2\sqrt{n}$