You will submit your solution to this assignment to the Curator System (as `HW1`). Your solution must be either a plain text file (e.g., NotePad) or a MS Word document; submissions in other formats will not be graded.

Except as noted, credit will only be given if you show relevant work.

---

1. [20 points] Using the rules given in the course notes, perform an exact count complexity analysis, for the worst case, of the body of the following function. (Take `list.length` to be N.)

```
int part(int[] list, int barrierIdx) {

   int barrier, maxIdx, temp;

   barrier = list[barrierIdx];                // 1
   maxIdx  = list.length - 1;                 // 2

   temp = list[barrierIdx];                   // 3
   list[barrierIdx] = list[maxIdx];           // 4
   list[maxIdx] = temp;                       // 5

   barrierIdx = 0;                            // 6

   for (int i = 0; i < maxIdx; i++) {         // 7
      if ( list[i] < barrier ) {              // 8
         temp = list[barrierIdx];             // 9
         list[barrierIdx] = list[i];          //10
         list[i] = temp;                      //11
         barrierIdx++;                        //12
      }
   }
   temp = list[maxIdx];                       //13
   list[maxIdx] = list[barrierIdx];           //14
   list[barrierIdx] = temp;                   //15

   return barrierIdx;                         //16
}
```

---

2. [40 points] For each part, determine the simplest possible function g(n) such that the given function is $\Theta(g)$. No justification is necessary.

a) $a(n) = 3n^2 + 14n + 47$

b) $b(n) = 14n + 3n \log n$

Hint: the last three take a little analysis.

c) $c(n) = n^{0.9} + \log n$

d) $d(n) = 3n^2 \log n + n^3$

e) $e(n) = 3n \log^2 n + 3n^2 \log n$

3. [20 points] Suppose that f and g are non-negative functions such that f is Θ(g). Is it necessarily true that:

$$2^{f(n)} \text{ is } \Theta\left(2^{g(n)}\right)$$

If so, prove it. (You may assume that the limit referred to in Theorem 8 exists.) If no, give a specific counter-example and show that it is a counter-example.

---

4. [20 points] Suppose that executing an algorithm on input of size N requires executing T(N) = N log N + 8N instructions. How long would it take to execute this algorithm on hardware capable of carrying out 2^{24} instructions per second if N = 2^{32}? (Give your answer in hours, minutes and seconds, to the nearest second.)