You may work in pairs for this assignment. If you choose to work with a partner, make sure only one of you submits a solution, and you paste a copy of the Partners Template that contains the names and PIDs of both students at the beginning of the file you submit.

You will submit your solution to this assignment to the Curator System (as `HW03`). Your solution must be either a plain text file (e.g., NotePad++) or a typed MS Word document; submissions in other formats will not be graded.

Partial credit will only be given if you show relevant work.

In all questions about complexity, functions are assumed to be nonnegative.

1. [36 points] The analysis of a certain algorithm leads to the following complexity function (for the average case):

$$T(N) = 3N\log^2(N) + 5N^2\log(N) + \log(N) + 17N + 2$$

Several computer science students offer their conclusions about the algorithm (quoted below). For each conclusion, state whether it is correct, or incorrect, or could be either correct or incorrect, based on the given information, and give a brief justification of your answer; feel free to cite any relevant theorems from the course notes.

a) The algorithm, on average, is $O(N^2)$.

b) The algorithm, on average, is $\Omega(N)$.

c) The algorithm, on average, is $\Omega(N^2 \log N)$.

d) The algorithm, on average, is $\Theta(N^2 \log N)$.

e) The algorithm, on average, is $O(N^3)$.

f) The algorithm, on average, is $\Theta(N \log^2 N)$.

g) In the worst case, the algorithm could be $\Theta(N^3)$.

h) In the worst case, the algorithm could be $\Omega(\log N)$.

i) In the worst case, the algorithm must be $\Omega(N)$.

j) The algorithm's best case performance is $\Omega(N^2 \log N)$.

k) The algorithm's best case performance cannot be $\Omega(N \log N)$.

l) The algorithm's best case performance is $O(\log N)$.

2. [24 points] Suppose that an algorithm takes 30 seconds for an input of 2^{24} elements (with some particular, but unspecified speed in instructions per second). Estimate how long the same algorithm, running on the same hardware, would take if the input contained 2^{30} elements, and that the algorithm's complexity function is:

a) $\Theta(N)$
b) $\Theta(\log N)$
c) $\Theta(N \log N)$
d) $\Theta(N^2)$

Assume that the low-order terms of the complexity functions are insignificant, and state your answers in the form HH:MM:SS.S (hours, minutes, seconds, tenths of a second). Be sure to show supporting work.

3. [18 points] Use theorems from the course notes to solve the following problems. Show work to support your conclusions.

a) Find the "simplest" function g(n) such that

$$f(n) = 17n^{3/2} + 3n \log n + 1000 \text{ is } \Theta(\text{g(n)})$$

b) Find the "simplest" function g(n) such that

$$f(n) = 5n \log^2 n + 8n^2 \log \text{n is } \Theta(\text{g(n)})$$

c) Find the "simplest" function g(n) such that

$$f(n) = \sqrt{n} + \log\left(\text{n+1000}\right) \text{ is } \Theta(\text{g(n)})$$

4. [12 points] Using the counting rules from the course notes, find the exact-count complexity function T(n) for the following algorithm. Show details of your analysis, and simplify your answer. In simplifying, you may discard the floor notation.

```
x = 100;
y = 0;
for (r = 1; r <= n; r++) {
   x = x + r;
   for (c = 2; c <= r; c = 2*c) {
      if ( x > y / c )
         y = y + r / c;
      else
         y = y - c;
   }
}
```

5. [10 points] Use the definition of Θ to prove that, if b is any positive constant, then

$$\text{log(n+b) is } \Theta\left(\log(n)\right)$$

Hint: you'll want to make use of several properties of the log() function.