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):

$$T(N) = 3N\log^2(N) + 5N^2\log(N) + 2N\log(N) + 23$$

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.

On average…

a) … the algorithm is $O(N^3)$.

b) … the algorithm is $\Omega(N^3)$.

c) … the algorithm is $\Omega(N^2 \log N)$.

In the worst case…

d) … the algorithm is $\Theta(N^2 \log N)$.

e) … the algorithm is $O(N^3)$.

f) … the algorithm is $\Omega(N^2 \log N)$.

In the best case…

g) … the algorithm is $O(N \log N)$.

h) … the algorithm is $O(N^2 \log(N))$.

i) … the algorithm is $\Omega(1)$.

2. [24 points] Suppose that an algorithm takes 30 seconds for an input of 2^{16} 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^{24} 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 "one-term" function g(n) such that

$$f(n) = 17n^{5/2} + 3n^2 \log n + 1000 \text{ is } \Theta(\text{g(n)})$$

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

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

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

$$f(n) = 3^n + n2^n \text{ is } \Theta(\text{g(n)})$$

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, if relevant.

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

5. [10 points] Use the <u>definition</u> of Θ to prove that, if *b* is an arbitrary positive integer, then

$$(n+1)^b \text{ is } \Theta\left(n^b\right)$$

Hint: One part is trivial. The other part is not trivial. You'll need to use some clever algebra. This result is also true if we drop the assumption that *b* is an integer; you might find this to be easier if you ignore that assumption as well.

Warning: a proof using any theorems, including the limit theorem, will receive NO credit.