Inlab8 – Big-O Notation

Objective

The objective of this in lab is to gain some practice with Big-O notation.

The definition of Big-O notation is as follows:
Suppose f and g are real-valued functions defined on the non-negative integers.
We say:

$$f \in O(g) \Leftrightarrow \exists c > 0, \text{ and } n_0 \geq 0 \text{ such that } \forall n \geq n_0 \ f(n) \leq cg(n)$$

In this exercise f will be the running time function $T(n)$ for the program or algorithm in question. The function g will be a simple function like n, n², n³, etc.

Part I
For each T(n) below, what is the Big-O complexity? You should provide the "tightest" simple function:

1. $T(n) = 10n + 100$
2. $T(n) = 3n^2 + 1$
3. $T(n) = n^4/100 + 9999n^3 - 100n + 2000$
4. $T(n) = 10$
5. $T(n) = n + 10\log(n)$

Part II
For each f and g, decide if $f \in O(g)$.
If you think the answer is true, indicate appropriate c and n_0 choices.

f	g
20n	n + 5
100	n + 5
3n³	10n² + n
$\log_2(n)$	n
100 n²	n³ / 100

What to Turn In

Turn in your written answers to the TA.