1. (10') Suppose \( f(n) = 8n + 16 \) and \( g(n) = 2n^2 + 3 \). Prove that \( f(n) \) is \( O(g(n)) \) by finding \( c > 0 \) such that \( f(n) \leq c \cdot g(n) \) for all \( n \).

2. (30') In each of the following situations, indicate whether \( f \) is \( o(g) \), or \( f \) is \( \omega(g) \), or \( f \) is \( \Theta(g) \) and explain or prove why you get the conclusion.

(a) \( n + 100 \quad n^2 + 1 \)
(b) \( 3n^{1/4} + 4 \quad 2n^{3/5} + 9 \)
(c) \( n \log n \quad 5n \log(5n) \)
(d) \( \log n \quad \log_2 n \)
(e) \( 100n + \log n \quad n + (\log n)^2 \)
(f) \( n^{1.03} \quad n(\log n)^2 \)
(g) \( \sqrt{n} \quad (\log n)^3 \)
(h) \( n^2 \quad 7^{\log n} \)
(i) \( n^2/\log n \quad n(\log n)^2 \)
(j) \( n^{2^n} \quad 5^n \)

Note: Assume base=2 for \( \log \) if not specified, i.e., \( \log n = \log_2 n \).

3. (30') Implement two programs \( pg1.c \) and \( pg2.c \) in C or C++. \( pg1.c \) uses function \( fib1() \) to calculate the Fibonacci numbers while \( pg2.c \) uses function \( fib2() \) to calculate the Fibonacci numbers.

Let each program try to calculate Fibonacci numbers \( F_0, F_1, F_2, F_3, \ldots \) until \( F_{64} \). (Note: “try” means that you can terminate your program when it takes more than 30 minutes to calculate a Fibonacci number.) It should print out the time so that you know how long it takes to calculate each Fibonacci number. The output looks like this:

```
Current time: hh:mm:ss
Fib(0) = 0
Current time: hh:mm:ss
Fib(1) = 1
Current time: hh:mm:ss
......
Fib(64) = ...
Current time: hh:mm:ss
```

You should submit following as your answers:

- For each program, describe what Fibonacci numbers take less than 1 second to calculate, what take 1 to 10 seconds, what take more than 10 seconds and up to 10 minutes, and what take more than 10 minutes.
- Email the two programs and their outputs as attachments to the TA/grader Weihua Liu (weihua.liu@uky.edu) and cc to the instructor (fei@cs.uky.edu) with subject “CS315 HW1”.

Hint: You may declare the type of resulting Fibonacci numbers as \textbf{long long int}. To get the time, you can use the following function as a reference.

```c
#include <stdio.h>
#include <time.h>
#include <stdlib.h>

#define TIME_H

void print_time()
{
    time_t now = time(NULL);
    struct tm* tm = localtime(&now);
    printf("The current time is: %d:%d:%d\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
}
```