Danville Programming Company
300 W. Roush Street
Danville, KE, 44444

April 8, 2009

Discrete Mathematicians
600 W. Walnut Street
Danville, KY 40244

Dear Discrete Mathematicians:

Good morning. I received your names from a friend of mine, O.N. Gutnic with the Danville Phone Company. I recently spoke with him about a problem I am having. He told me that your mathematical cleverness and aptitude could help.

I am currently working on a programming assignment for my internship with The Danville Programming Company. During my programming project, my algorithm must evaluate a polynomial. I believe the conventional algorithm for evaluating a polynomial $a_n x^n + a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \ldots + a_0$ at $x=c$ can be expressed in pseudo-code by:

Input: $c, a_0, a_1, a_2, a_3, \ldots, a_n$ [all real numbers]

Algorithm Body:

```
power := 1
y:= a0
for i:= 1 to n
    power:= power • c
    y:= y + ai • power
next i
```

Output: $y = a_n c^n + a_{n-1}c^{n-1} + a_{n-2}c^{n-2} + \ldots + a_0$

I have written the program in pseudo-code to improve its readability.

First, I am trying to determine if this algorithm will in fact evaluate the polynomial at some constant c. Then, if the code is correct, how long will it take for this algorithm segment to be executed. Here at the Danville Programming Company, we assume that each operation, on average, takes about one nanosecond to execute. In your opinion, is this algorithm efficient?

Typically, for this particular program, the polynomials I will be evaluating are *extremely* large. If my polynomial has a degree of 1 trillion, how many hours will it take for this program to be executed? What about for any general value of *n*.

I would be very grateful if you would submit the full details on your efforts in a typed response so that, in the future, I will hopefully be able to solve similar problems. I would appreciate an answer as soon as possible, but certainly no later than April 24th.

Yours Sincerely,

O. Tinoaton
Danville Programming Company