Laboratory 6

1. Use ifactor to factor your SSN, telephone number and other fancy numbers. Find out for what number it takes the largest time to factor.

2. Use ifactor to factor 123,1234567,123456789,1234567891011,....etc. Can you make a conjecture that all these numbers are composite? Can you justify this conjecture?

3. Implement the following algorithm in Maple. This algorithm factors **LARGE** numbers.

   Algorithm A (Factoring By Trial Division)
   A1[Initialize]: $t = 0$; $k = 0$; $n = N$; (N is the number to be factored, d_k is the $(k+1)$th prime, $d_0 = 2$, p_t will be the t-th factor, and during this algorithm the variables t, k, n are related by the following condition: $n = N/(p_1...p_t)$ and n has no prime factors less than d_k.)
   A2[n=1?]. If $n = 1$, the algorithm terminates.
   A3[Divide]. $q = \text{floor}(n/d_k)$; $r = n \bmod d_k$. (Here q and r are the quotient and reminder obtained when n is divided by d_k.)
   A4[Zero remainder?] if $r \neq 0$, go to A6.
   A5[Factor found]. $t = t + 1$; $p_t = d_k; n = q$; return to step A2.
   A6[Low quotient?] if $q > d_k$, then $k = k + 1$ and go to step A3.
   A7.[n is prime]. $t = t + 1$. $p_t = n$ and terminate the algorithm.

4. Compare the performance of this program with ifactor by testing for numbers in 1 and 2. Could you find example in which your program ran faster?