MATH 471: HOMEWORK 1
“TAYLOR SERIES AND COMPUTER ARITHMETIC”
SUMMER 2003

NOTE: For each homework assignment please follow the “Guidelines for Numerical Analysis HW”. For any problems calling for a MATLAB function or script, you must submit these, with a comment on top including your name, the date, and “MATH 471”. And please clearly label any and all plots (title, x-label, y-label, and legend).

1. (a) Construct the 1st, 2nd and 3rd order Taylor polynomials and the associated remainder term for the function \( f(x) = \sqrt{2x - x^2} \) about the point \( x = 1 \) for \( x \in (0, 2) \)

(b) Make a single MATLAB plot which contains \( f(x) \), \( p_1(x) \), \( p_2(x) \), and \( p_3(x) \) all plotted versus \( x \) on the given interval.

2. The Taylor polynomial \( p_n \) for function \( f(x) = e^x \) expanded about the point \( x_0 = 0 \) has remainder term

\[
R_n(x) = \frac{1}{(n + 1)!} e^c x^{n+1}
\]

for some \( c \) between 0 and \( x \). Using this remainder term, determine the smallest value of \( n \) needed to guarantee that the absolute error \( |p_n(1) - f(1)| < 10^{-5} \).

3. Consider the function

\[
g(x) = \frac{\log(1 + xe^x)}{x}
\]

which is undefined at \( x = 0 \).

(a) Approximate the numerator of \( g \), \( \log(1 + xe^x) \), by a Taylor polynomial of degree 2 about the point \( x = 0 \).

(b) Use this Taylor approximation to determine \( \lim_{x \to 0} g(x) \).

4. Convert the following numbers from binary to decimal. SHOW ALL WORK.

(a) 110101

(b) 110.101
5. The following algorithm
   \[ x_0 := x; \; j := 0 \]
   while \( x_j \neq 0 \), do
   \[ a_j := \text{remainder of integer divide } x_j/2 \]
   \[ x_{j+1} := \text{quotient of integer divide } x_j/2 \]
   \[ j := j + 1 \]
   end while

can be used to convert a positive decimal integer \( x \) to its binary equivalent,
\[ x = (a_n a_{n-1} \cdots a_1 a_0)_{2}. \]

Implement this algorithm in MATLAB and apply it to convert the following integers to their binary equivalents.

(a) 45
(b) 2904

**HINT:** The MATLAB functions \texttt{rem, mod} and \texttt{floor} might be helpful. Try “help \texttt{rem}” etc. to see how to use them.

6. Consider
\[ x_n = 1 - \cos \left( \frac{\pi}{n} \right) \]
\[ y_n = 2 \sin^2 \left( \frac{\pi}{2n} \right) \]

**NOTE:** From trigonometry we know that \( x_n = y_n \) since \( \sin^2(\theta) = \frac{1}{2} (1 - \cos(2\theta)) \).

(a) subplot \#1: Plot \( x_n \) and \( y_n \) versus \( n \) for \( n = 1, \ldots, 1000 \) on a single log-log plot using MATLAB’s \texttt{loglog} command. Use a solid line for \( x_n \) and open circles for \( y_n \).

(b) subplot \#2: Plot the relative error \( |x_n - y_n|/|y_n| \) on a log-log plot using the \texttt{loglog} command.

(c) As \( n \) becomes large, which one (\( x_n \) or \( y_n \)) is more accurate? Why?