MTH 308: Numerical Analysis & Scientific Computing I
Department of Mathematics and Statistics
Indian Institute of Technology - Kanpur March 28, 2019

Lab 10

Steps you should follow :

1. Start a Matlab session by typing

   ```
   $ matlab &
   ```

   at the command prompt.

2. In the Matlab command window, type

   ```
   >> diary lab10_ID
   ```

   where ID stands for your roll number. For example, if your roll number is 12345, then the command will be `diary lab10_12345`; if your roll number is 123456, then the command will be `diary lab10_123456`. This will create a file named `lab10_ID` in the present working directory. PLEASE DO NOT EDIT THIS FILE.

3. Do your lab assignment – create a separate file to write your scripts/functions, if required; once done, at the Matlab's command prompt, type

   ```
   >> diary off
   ```

4. Attach this file (that is, `lab10_ID`) and any other Matlab code file that you may have created for the labwork in an **email with subject Lab10-ID** and send it to

   mth308.iitk@gmail.com

   before the end of the lab session, that is, by 4:30 pm. Note that **late submissions will not get any credit**. In the case that your diary file is too big to be sent as an email attachment, upload it on you Google Drive (the cloud storage space associated with your Google/Gmail account) and share the link with mth308.iitk@gmail.com.

We have seen that using computer arithmetic, we can evaluate polynomials exactly (up to the rounding error). Let the polynomial $p(x)$ of degree n be given by

$$p(x) = \sum_{k=0}^{n} a_k x^k. \tag{1}$$

Note that $p(x)$ can be rewritten as

$$p(x) = a_0 + x(a_1 + x(a_2 + x(a_3 + \cdots + x(a_{n-1} + xa_n)\cdots))). \tag{2}$$

Explain why using the expression in (2) is more efficient over the expression in (1) for evaluating the polynomial. Write a Matlab program for implementing the polynomial evaluation using (2) and use it to compute π with the help of

$$p_n(x) = \sum_{k=0}^{n} (-1)^k \frac{x^{2k+1}}{2k+1}, \qquad |x| \leq 1$$

the nth order Taylor series approximation to $\tan^{-1} x$. Complete the following table

n	$\lvert\pi - 6p_n(1/\sqrt{3})\rvert/\pi$	$\lvert\pi - 4p_n(1)\rvert/\pi$
8	–	–
16	–	–
32	–	–
64	–	–
128	–	–
256	–	–
512	–	–
1024	–	–
2048	–	–
4096	–	–

Note that your Matlab script, upon running, should print/display/output this table as an array. Comment on the result.