M.Sc. in High-Performance Computing
5633 - Numerical Methods
Assignment 1

Marina Krstic Marinkovic
(mmarina@maths.tcd.ie)
School of Mathematics, TCD

Rules

To submit, make a single tar-ball with all your code and a pdf of any written part you want to include. Submit this via `msc.tchpc.tcd.ie` or via email to `mmarina@maths.tcd.ie` by the end of **Friday November 2nd, 2018**. Before uploading the solutions, please rename your tarball and the corresponding folder using the following convention: `5633_hw1_{uname}.tar`. Instead of R, you may use Matlab or Python for the numerical/plotting part. Late submissions without prior arrangement or a valid explanation will result in reduced marks.

Question

1. (10p) Convert the following nubers into IEEE 754 single precision numbers:

   (a) -55.0 **[4p]**

   (b) 0.55 **[6p]**

   and then use their binary representation to write these numbers in hexadecimal representation. The conversion table between decimal, binary and hexadecimal representation is given in Figure 1.

2. (20p) Write a R script that uses the bisection method to find the root of a given function on a given interval, and apply this program to find the roots of the functions bellow on the indicated intervals. Be careful to avoid unnecessary function calls in a single iteration. Use a relationship derived in class to determine *a priori* the number of steps necessary for the root to be accurate within 10^{-8} for the functions:

   (a) $x^3 - 2$, $[a, b] = [0, 2]$, **[3p]**

   (b) $e^x - 2$, $[a, b] = [0, 1]$. **[3p]**

Binary	Hex	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Figure 1: Conversion table between decimal, binary and hexadecimal representation.

Apply Newton's method to find the root of the second function, $f(x) = e^x - 2$, for which the exact solution is $\alpha = \ln 2$. **[4p]**

Perform the following experiments:

(c) Compute the ratio

$$R_n = \frac{\alpha - x_{n+1}}{(\alpha - x_n)^2}$$

and observe whether or not it converges to the correct value as $n \to \infty$. **[3p]**

(d) Compute the modified ratio

$$R_n = \frac{\alpha - x_{n+1}}{(\alpha - x_n)^p},$$

for various $p \neq 2$, but near 2. What happens? Comment, in the light of the definition of order of convergence. **[2p]**

Apply the secant method to the same function, using x_0, x_1 equal to the end points of the given interval. Stop the iteration when the error as estimated by $|x_n - x_{n-1}|$ is less than 10^{-8}. Compare to your results for Newton and bisection method. **[5p]**

Have your codes print out the entire sequence of iterates to the following files `bisection1.txt` (for function a)), `bisection2.txt`, `newton.txt`, and `secant.txt` (for function b)). In the case of bisection method, the file should have following five columns:

$$n \quad x_n \quad b_{n-1} - a_{n-1} \quad a_n \quad b_n,$$

the file for the Newton's method should contain the following four columns:

$$n \quad x_n \quad \alpha - x_n \quad log_{10}(\alpha - x_n),$$

and the file for the secant method should contain the following three columns:

$$n \quad x_n \quad \alpha - x_n.$$

3. (20p) Write an R code that performs

   - LU factorisation without partial pivoting **[4p]**
   - LU factorisation with partial pivoting **[6p]**

   for an arbitrary $n \times n$ matrix (n<100). Use it to solve the systems of equations $Ax = b$:

   (a) $A = H_5$,
   $b = [5.0, 3.550, 2.81428571428571, 2.34642857142857, 2.01746031746032]^T$; **[4p]**

   (b) $A = A_5, \qquad b = [-4, -7, -6, -5, 16]^T$, **[4p]**

   where H_n and A_n denote the families of matrices parameterised by their dimension in the following way:

$$\begin{aligned} H_n &= [h_{ij}], \quad h_{ij} = \frac{1}{i+j-1}, \\ A_n &= [a_{ij}], \quad a_{ij} = \begin{cases} 1, & i = j; \\ 4, & i - j = 1; \\ -4, & i - j = -1; \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

   In each case, have your code multiply out the L and U factors to check that the routine is working properly. **[2p]**