Lab Session 2

MA-423 : Matrix Computations Lab 2017 R. Alam

---

1. The roots of a quadratic polynomial $p(x) := ax^2 + bx + c$ is given by $(-b \pm \sqrt{b^2 - 4ac})/2a$. Write a MATLAB function that implements the above formula to compute the roots. Your function will look like this:

```
function [x1, x2] = quadroot1( a, b, c)
d = sqrt( b^2 - 4 * a* c );
x1 = (-b + d) / (2*a);
x2 = (-b - d) / (2*a);
```

   The largest (in magnitude) root of p can be computed as $x_1 = (-b - \text{sign}(b)\sqrt{b^2 - 4ac})/2a$ and the second root x_2 can be computed from the identity $x_1 x_2 = c/a$. Write a MATLAB function that implements the modified method to compute the roots. Your function will look like this:

```
function [x1, x2] = quadroot2(a, b, c)
d = sqrt( b^2 - 4 * a* c );
x1 = (-b - sign(b) * d) / (2*a);
x2 = c/( a * x1 );
```

   Find the roots of $x^2 - (10^7 + 10^{-7})x + 1)$ using `quadroot1` and `quadroot2`. Do you observe any difference? Which method is better and why?

2. The machine epsilon `eps` of a floating point system is the distance from 1 to the next floating number bigger than 1 and `u = eps/2` is the unit roundoff (default). You can compute `eps` and `u` in MATLAB by writing a small script. What is it that the following MATLAB script computes?

```
x = 2;
while x > 1
x = x/2
end
```

   On the other hand, if the condition $x > 1$ in the above script is replaced with $1 + x > 1$ then what will be the output?

3. This exercise illustrates the difficulty in handling polynomials in finite precision computation. Consider the polynomial

$$\begin{aligned} p(x) = (x-2)^9 &= x^9 - 18x^8 + 144x^7 - 672x^6 + 2016x^5 \\ &\quad -4032x^4 + 5376x^3 - 4608x^2 + 2304x - 512 \end{aligned}$$

   Write a MATLAB script to evaluate p at 151 equidistant points (use `linspace` command) in the interval [1.95, 2.05] using two methods:

(a) Apply Horner's method, or call MATLAB function `polyval` (Type `help polyval` for more info).

(b) Calculate $p(x) = (x-2)^9$ directly.

Plot these results in two separate figures. For example, if x is a row vector of points in the given interval then the commands

```
>> y = p(x); plot(x, y)
```

will do the job. Do the plots differ from one another? If yes, can you think of possible reasons?

**********End**********