PW 02: SOLVING A NONLINEAR EQUATIONS f(x)=0

I. THE BISECTION ALGORITHM

This simple algorithm assumes that an initial interval is known in which a root of the equation **f(x)=0** lies and then proceeds to reduce this interval until the required accuracy is achieved for the root. This algorithm is mentioned only briefly since it is not in practice used by itself but in conjunction with other algorithms to improve their reliability. The algorithm may be described by:

Slove **f(x)=0** by bisection	
1	find an interval **[a b]** in which **f(x)=0**
2	set **c** to **(a+b)/2**
3	calculate **f(c)**
4	test for convergence
5	if **f(c)** and **f(b)** have opposite sign set **a** to **c**, otherwise set **b** to **c**
6	repeat from step 2

II. NEWTON–RAPHSON METHOD

The Newton–Raphson method pursues the idea by allowing the two points of the secant method to merge into one. In so doing it uses a single point and the tangent to the function at that point to construct a new estimate to the root.

Slove **f(x)=0** by the Newton–Raphson method	
1	choose a starting point **a**
2	if **f '(a)≠0 replace a by a-f(a)/ f '(a)** otherwise
3	restart using a different **a**
4	test the convergence
5	repeat from step 2

III. SUCCESSIVE APPROXIMATIONS

Slove **f(x)=0** by the successive approximations method	
1	The equation must be rewritten **f(x)=0** in the form **x=g(x).**
2	The sufficient but not necessary convergence condition is: $\mathbf{|g(x)|<1}$ everything $\mathbf{x \in [a\ b]}$
3	To approximate the solution of the equation, we start from the initial value $\mathbf{x_0}$=...
4	We iteratively calculate the values of $\mathbf{x_n}$ with : $\mathbf{x_n=g(x_{n-1})}$
5	The stopping criteria can be: $\mathbf{|x_n - x_{n-1}|<\varepsilon}$

IV. CONTROL STRUCTURES

We recall the syntax of control and repetition structures:

IV.1. The if test statement

if *condition1 action1*

elseif *condition2 action2*

...

else *action3*

end

IV.2. The for loop

for *counter = initial_value : increment : final_value*

action

end

IV.3. The while loop

while *condition*

action

end

V MANIPULATIONS

Let's solve the equation $\mathbf{f(x)=1-x.e^x=0}$ where $\mathbf{x \in]0\ \infty]}$.

1. Form a list of values (a vector) **x** ranging from **0** to **1** with a step of **0.01** ;
2. use **inline** command to declare function **f(x)**;
3. Plot the graph y=**f(x)** on an interval **[0 1]**;
4. Write a script, which you will call "***bisection.m***" that implements the bisection method by following these steps:

- Initialize the limits of the search domain **a** and **b**;
- Initialize an iteration counter **k** to **0**;
- Write the bisection algorithm by incrementing the counter **k** at each passage of the **while** loop;
- Stop the loop when the width of the domain becomes **$<=10^{-3}$**;
- Display the calculated solution as well as the number of iterations.
- use Matlab's predefined command **fzero**, to find the solution to the equation **f(x)=0**.

5. Write another script, which you will call "**Newton.m**" that implements the Newton-Raphson method.
6. Write another script, which you will call "**Successive.m**" that implements the successive approximation method. To do this, we must rewrite the equation in the form **x=g(x)** while ensuring the convergence of the algorithm.
7. Compare the different methods implemented.