

Numerical Computation HW₁

Sina Barazandeh , VahidReza Niazmand

26 April 2019

1 False Position method :

Matlab implementation of this method is attached. This method is a very old method for solving an equation in one unknown, that, in modified form, is still in use. In simple terms, the method is the trial and error technique of using test ("false") values for the variable and then adjusting the test value according to the outcome.

This program gets a function as its input, then 2 initial variables to start the calculations and a tolerance meaning of allowed error.

```
for i=0:inf
    x2= x1 - (f(x1)* (x1-x0)/(f(x1)-f(x0)))
    c = f(x2)
    absolute_c= abs(c);
    if absolute_c < tolerance
        break
    end
    if f(x0)*c <0
        x1=x2;
        continue
    else
        x0=x2;
        continue
    end
end
```

As in implementation above, this process is repeated until reaches an acceptable error(here variable 'c'). Here is an example of an input and the calculated output.

input :

$$f(x) = x^3 - 7 \quad (1)$$

initial inputs : 1 , 3 considering that we guess that our answer is somewhere between 1 and 3 and a tolerance of 0.1 meaning that 0.1 error would cause no problems. after 5 times repeating the loop the result is:

output :

$$root = 1.9072, error = -0.0629 \quad (2)$$

this result means that the answer(root) of the given function is approximately 1.9072 and for this value(1.9072) the error is about 0.07 which is less than allowed error(here it is 0.1).