

---

## Méthodes itératives

---

### 1. Correction de l'exercice 1 (algorithme de Newton) en SCILAB

```
//newton pour resoudre x^2-M=0 (recherche de racine carree)
//NB : pour afficher plus de chiffres, faire format(20) (par exemple)
function res=NewtonRacineCarree(a,x0,tol)
  itermax=20;//pour eviter des boucles infinies
  x=x0;//initialisation
  iter=1;
  test=1;
  while (test>tol)&(iter<itermax)
    y=x/2+a/(2*x);
    test=abs(y-x);
    x=y;
    disp(x)
    iter=iter+1;
  end
  res=x;
endfunction
```

Compiler la fonction. La sortie en ligne donne :

```
-->format(20)

-->NewtonRacineCarree(4,2.5,1e-16)

2.04999999999999982

2.00060975609756087

2.00000009292229475

2.000000000000000222

2.

2.
ans =

2.
```

---

### en MAPLE

```
> NewtonRC:=proc(a,x0,tol)
> local k,x,y,test;
> x:=x0;test:=1;
```

```

> while test>tol do
>
>     y:=evalf(x-(x*x-a)/(2*x));
>     print(x);
>     test:=abs(y-x);
>     x:=y;
> end do;
> end proc;
    NewtonRC := proc(a, x0, tol)
    local k, x, y, test;
      x := x0;
      test := 1;
      while tol < test do y := evalf(x - (x*x - a)/(2*x));
        print(x);
        test := abs(y - x);
        x := y;
      end do;
    end proc;
> Digits:=20;NewtonRC(4,4,0.001);
                Digits := 20
                    4
                2.50000000000000000000
                2.05000000000000000000
                2.0006097560975609756
                2.0000000929222946603

```