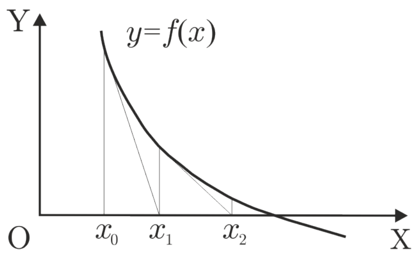
PRACTICE 2

1 Newton-Raphson's method

We use this method when the exact differential may be computed.

- Start with x_0 close to the root r, define x_1 to be the point of intersection of the x-axis and the tangent line to the curve at the point $(x_0, f(x_0))$.
- The process is repeated to obtain a sequence $\{x_n\}$ that converges to r.



The line tangent to the curve at the point $(x_0, f(x_0))$ is $y - y_0 = m(x - x_0)$, thus:

$$y - f(x_0) = f'(x_0)(x - x_0)$$

and the intersection with the x-axis ($y = 0$) is

$$-f(x_0) = f'(x_0)(x - x_0) \implies -\frac{f(x_0)}{f'(x_0)} = x - x_0 \implies x_0 - \frac{f(x_0)}{f'(x_0)} = x$$

therefore,

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \quad \implies \cdots \implies \quad x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

The sequence $\{x_n\}$ that converges to r when $\frac{|x_{n+1}-x_n|}{|x_n|}$ becomes very small, thus the stopping criterion is:

$$\frac{|x_{n+1} - x_n|}{|x_n|} \leq T$$

being T the relative error (known as tolerance). To prevent possible divisions by numbers close to zero, we will raise it as $|x_{n+1} - x_n| \leq T \cdot |x_n|$.

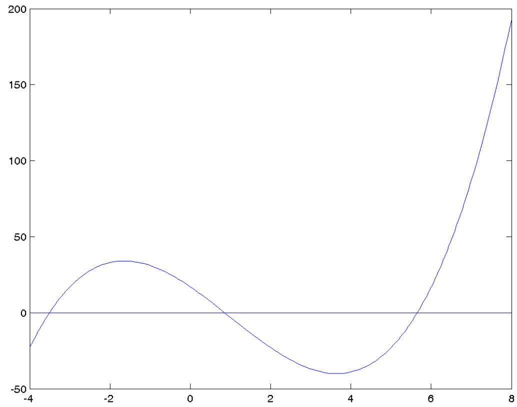
Example 1.1 *Check graphically that the equation* $x^3 - 3x^2 - 18x + 17 = 0$ *has its three solutions at the interval* $[-4, 8]$. *Find them by applying the Newton-Raphson method, with relative tolerance* 10^{-6}.

Solution: We define $f(x) = x^3 - 3x^2 - 18x + 17$ as symbolic function, so we can obtain the derivative with MatLab

```
>> syms x; f(x)=x^3-3*x^2-18*x+17
f(x) =
x^3 - 3*x^2 - 18*x + 17
>> fd=diff(f)
fd(x) =
3*x^2 - 6*x - 18
```

and we convert them to numerical functions and plot the graph of $y = f(x)$:

```
>> f_num=matlabFunction(f)
f_num =
@(x)x.*-1.8e1-x.^2.*3.0+x.^3+1.7e1
>> fd_num=matlabFunction(fd)
fd_num =
@(x)x.*-6.0+x.^2.*3.0-1.8e1
>> fplot(f_num,[-4 8]);
>> hold on; fplot(@(x) 0,[-4 8]);
```



Finally, we apply the **newton** function, and the three solutions are:

```
>> x1=newton(f_num,fd_num,-3,1e-6,200)
x1 =
-3.5094
>> x2=newton(f_num,fd_num,2,1e-6,200)
x2 =
0.8570
>> x3=newton(f_num,fd_num,6,1e-6,200)
x3 =
5.6524
```

Example 1.2 *Create the function* `newton2` *by changing the function* `newton`*, so that the output variable contains the values of the iterations of the Newton-Raphson method. Then, apply* `newton2`*, for solving the equation of the previous example.*

Solution: We create the file newton2.m with the following lines,

and we find the solutions of the equation of the previous example,

```
>> y1=newton2(f_num,fd_num,-3,1e-6,200)
y1 =
-3.6296    -3.5140    -3.5094    -3.5094    -3.5094
>> y2=newton2(f_num,fd_num,2,1e-6,200)
y2 =
```

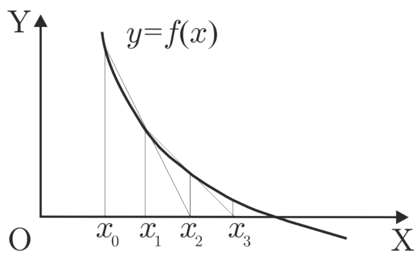
```
0.7222    0.8576    0.8570    0.8570
>> y3=newton2(f_num,fd_num,6,1e-6,200)
y3 =
5.6852    5.6527    5.6524    5.6524
```

2 Secant method

The Newton-Raphson algorithm requires the evaluation of $f(x)$ and $f'(x)$ and, sometimes, it is desirable to have a method that converges almost as fast as Newton's method yet only involves evaluations of $f(x)$ and not of $f'(x)$. The secant method will only require one evaluation of $f(x)$ per step and it is almost as fast as Newton's method:

- two initial points x_0, x_1 near the root are needed.
- the intersection of the secant line through the points $(x_0, f(x_0))$ and $(x_1, f(x_1))$, and the ax Ox (the value of x such that y=0) gives us x which will be used as x_2 in the next step.

This process is shown in the following graph:



The secant line through the points $(x_0, f(x_0))$ and $(x_1, f(x_1))$ is:

$$\frac{y - f(x_1)}{x - x_1} = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

The intersection wit Ox (the value of x such that y=0) is: $-\dfrac{f(x_1)}{x - x_1} = \dfrac{f(x_1) - f(x_0)}{x_1 - x_0} \implies$

$$\implies -f(x_1)\frac{x_1 - x_0}{f(x_1) - f(x_0)} = x - x_1 \implies x_1 - f(x_1)\frac{x_1 - x_0}{f(x_1) - f(x_0)} = x$$

thus,

$$x_2 = x_1 - f(x_1)\frac{x_1 - x_0}{f(x_1) - f(x_0)} \implies \cdots \implies x_{n+2} = x_{n+1} - f(x_{n+1})\frac{x_{n+1} - x_n}{f(x_{n+1}) - f(x_n)}$$

And the stopping criterion is $|x_{n+2} - x_{n+1}| \leq T \cdot |x_{n+1}|$.

Example 2.1 *Find again the solutions of the equation* $x^3 - 3x^2 - 18x + 17 = 0$ *by applying the secant method with the relative tolerance* 10^{-6}.

Solution: we find the three solutions as follows:

```
>> z1=secant(f_num,-4,-2,1e-6,200)
z1 =
   -3.5094
>> z2=secant(f_num,0,2,1e-6,200)
z2 =
    0.8570
>> z3=secant(f_num,4,6,1e-6,200)
z3 =
    5.6524
```

3 Fixed point iteration method

This method is used to solve equations in the form $g(x) = x$, that is, geometrically, the intersection of the curve $y = g(x)$ with the bisector of the first quadrant $y = x$.

The idea is, from a starting value x_0, we construct the sequence $x_{n+1} = g(x_n),\ n \in \mathbb{N}$; if the assumptions of the fixed point theorem are fulfilled, this sequence converges to the desired solution, regardless of the choice made of the starting value x_0 (sometimes known as "seed").

So, we construct the sequence and we use the following as a stopping criterion

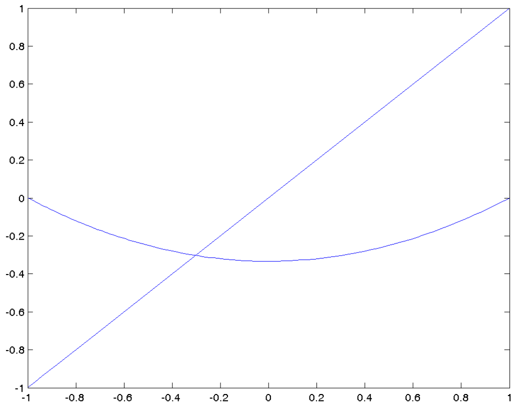
$$\frac{|g(x_n) - x_n|}{|x_n|} = \frac{|x_{n+1} - x_n|}{|x_n|} \leq T$$

being T the relative error we want to use (known as tolerance). To prevent possible divisions by numbers close to zero, we will raise it as $|x_{n+1} - x_n| \leq T \cdot |x_n|$ and, thus, a way of programming this method is to create the file fixpt.m as follows

Example 3.1 *Check graphically that the function* $g(x) = \dfrac{x^2 - 1}{3}$ *has an unique fixed point in the interval* $[-1, 1]$ *and find it using the starting value* $x_0 = 0$ *and the relative tolerance* 10^{-6}.

Solution: We define the function $g(x)$. We produce a graph of the function $y = g(x)$ and the line $y = x$ in the interval $[-1, 1]$,

```
>> g=@(x) (x^2-1)/3
g =
@(x)(x^2-1)/3
>> fplot(g,[-1 1]);
>> hold on; fplot(@(x) x,[-1 1]);
```



we see that the fixed point is the intersection of both functions and there is only one point. We can find the fixed point by typing:

```
>> x=fixpt(g,0,1e-6,100)
x =
-0.3028
```

Example 3.2 *Create a function in file* `fixpt2`*, by modifying the function* `fixpt`*, so that the output argument is a vector whose elements are the sequence of the fixed point iteration method. Then, apply* `fixpt` *and* `fixpt2` *to the function*

$$g(x) = \sqrt{\frac{x+3-x^4}{2}}$$

with starting value $x_0 = 1$, relative tolerance 10^{-6} and maximum number of iterations 100.

and we check it with the same function as in the previous example

```
>> x=fixpt2(g,0,1e-6,100)
x =
Columns 1 through 7
-0.3333   -0.2963   -0.3041   -0.3025   -0.3028   -0.3028   -0.3028
Columns 8 through 10
-0.3028   -0.3028   -0.3028
```

Now, we define the function $g(x)$

```
>> g=@(x) sqrt((x+3-x^4)/2)
g =
@(x)sqrt((x+3-x^4)/2)
```

and we apply both functions to it

```
>> x=fixpt(g,1,1e-6,100)
It does not converge with the requested accuracy.
The result is not the fixed point but the last iteration.
```

```
x =

0.9306

>> x=fixpt2(g,1,1e-6,100);
It does not converge with the requested accuracy.
```

we see that with 100 iterations we don't find the fixed point, so we wonder if we are considering enough iterations. However, if we look at the last six elements of the sequence

```
>> x(95:end)
ans =
1.2611    0.9306    1.2611    0.9306    1.2611    0.9306
```

It is clear that it is not worth to increase the iterations, since the sequence is not convergent (the subsequence of even elements has different limit than the subsequence of odd elements).

4 The fzero command

The fzero command is an M-file. The algorithm, which was originated by T. Dekker, uses a combination of bisection, secant, and inverse quadratic interpolation methods to solve the equation $f(x) = 0$.

The syntax:
`x=fzero(f,x0)` tries to find a zero of `f` near `x0`, if `x0` is a scalar and `f` is an anonymous function. If `f` is a function defined by a file then the syntax is `x=fzero(@f,x0)`.

`[a,b]` is an interval where the sign of `f(a)` differs from the sign of `f(b)`. An error occurs if this is not true. Calling fzero with such an interval (`fzero(f,[a,b])` o `fzero(@f,[a,b])`) guarantees fzero will return a value near a point where `f` changes sign.

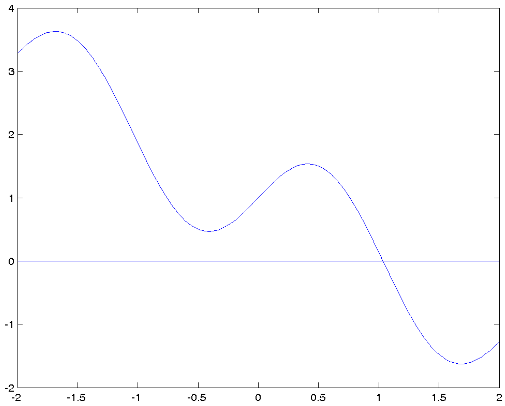
Example 4.1 *Solve the equation* $\sin(3x) - x + 1 = 0$*, using the* **fzero** *command.*

Solution: If x is the a root of the equation, it must verify

$$x = \sin 3x + 1 \implies |x| \leq |\sin 3x| + 1 \leq 2$$

thus, $x \in [-2, 2]$. Therefore, we can plot the function $\sin(3x) - x + 1$ in this interval

```
>> f=@(x) sin(3*x)-x+1
f =
    @(x)sin(3*x)-x+1
>> fplot(f,[-2 2]);
>> hold on; fplot(@(x) 0,[-2 2]);
```



there is a solution close to 1, so we can find it executing

```
>> fzero(f,1)
ans =
    1.0354
```

or it could also be solved through:

```
>> fzero(f,[0.5 1.5])
ans =
    1.0354
```