Exercício 4

(a) Estime quantas iterações do método da Bisseção serão necessárias para determinar $\sqrt{3}$ com precisão 10^{-4} no intervalo $[1; 2]$.

In [1]:
```python
import math
```

In [2]:
```python
def fx(x):
    return x**2 -3
```

In [3]:
```python
a=1
b=2
precisao = 10**-4
k = (math.log(b-a,10) - math.log(precisao,10)) / math.log(2,10)
k
```

Out[3]:
```
13.287712379549449
```

In [4]:
```python
x = a
k = 20
it = 0

while True:
    it +=1
    #print("x= ", x,"iteracoes =", it,"f(x) = ",fx(x))
    xOld = x
    x=a+((b-a)/2)

    if fx(a)*fx(x) < 0:
        #print("x=b= ", x)
        b=x
    else:
        #print("x=a= ", x)
        a=x
    if k == it:
        break


print("x= ", x,"iteracoes =", it,"f(x) = ",fx(x))
```

```
x=  1.732050895690918 iteracoes = 20 f(x) =  3.052637111977674e-07
```

Resposta a:

$k \approx 14$
O resultado encontrado foi de $x \approx 1.73199$ para $y \approx -0.00019$

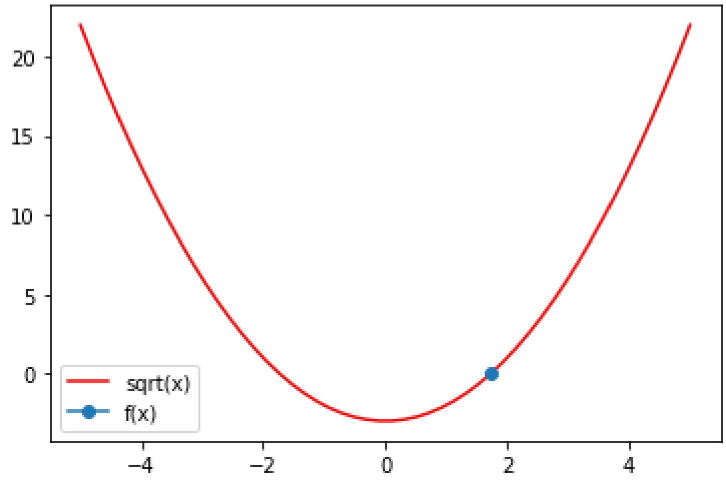
(b) Determine um valor aproximado para $\sqrt{3}$ com precisão 10^{-4} utilizando o algoritimo da bisseção.

Resposta b:

In [5]:
```python
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
```

```
xx = np.linspace(-5,5,50)
plt.figure(1)
y= xx**2 -3
plt.plot(xx,y,'-r',label = 'sqrt(x)')
plt.plot(1.73206,0,'-',label = 'f(x)',marker = 'o')
plt.legend()
```

Out[5]: <matplotlib.legend.Legend at 0x20028a7fd90>



In [6]:

```
a=1
b=2
x = a
k = 14
it = 0

while True:
    it +=1
    xOld = x
    x=a+((b-a)/2)

    if fx(a)*fx(x) < 0:
        b=x
    else:
        a=x

    Er = abs((x-xOld)/x)
    if Er < precisao:
        break

print("xOld= ", xOld,"x= ", x,"iteracoes =", it,"f(x) = ",fx(x))
```

```
xOld=  1.732177734375 x=  1.7320556640625 iteracoes = 13 f(x) =  1.6823410987854004e
-05
```

O resultado encontrado foi de $x \approx 1.73206$ para $y \approx -0.00001$

(c) Compare o número de iterações nos itens (a) e (b) deste exercício. Dê sua explicação pa

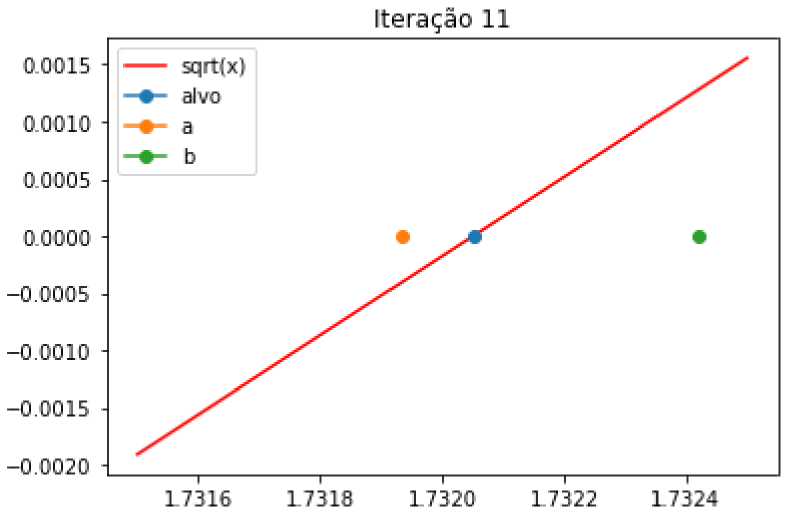
Resposta c:

In [7]:

```
xx = np.linspace(1.7315,1.7325,1000)
plt.figure(1)
y= xx**2 -3
plt.plot(xx,y,'-r',label = 'sqrt(x)')
plt.plot(1.7320508075688772,0,'-',label = 'alvo',marker = 'o')
plt.plot(1.731933,0,'-',label = 'a',marker = 'o')
plt.plot(1.732421,0,'-',label = 'b',marker = 'o')
```

```
plt.title("Iteração 11")
plt.legend()
```

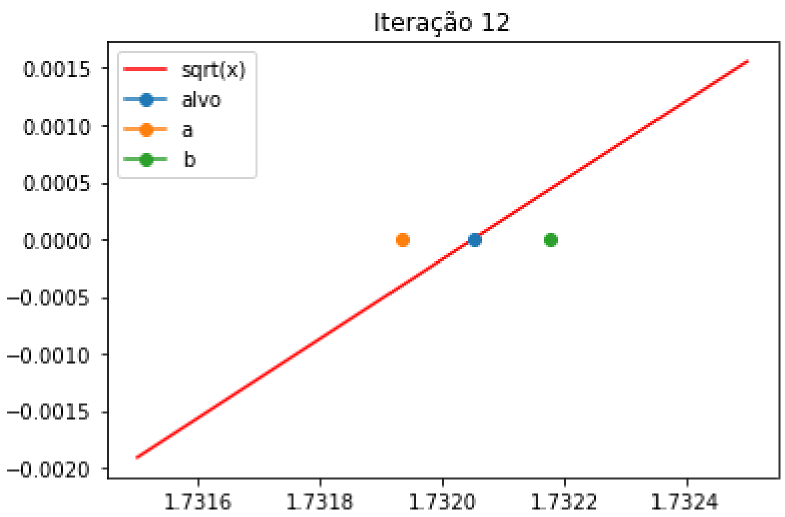
Out[7]: <matplotlib.legend.Legend at 0x200268f4df0>



In [8]:

```
xx = np.linspace(1.7315,1.7325,1000)
plt.figure(1)
y= xx**2 -3
plt.plot(xx,y,'-r',label = 'sqrt(x)')
plt.plot(1.7320508075688772,0,'-',label = 'alvo',marker = 'o')
plt.plot(1.731933,0,'-',label = 'a',marker = 'o')
plt.plot(1.732177,0,'-',label = 'b',marker = 'o')
plt.title("Iteração 12")
plt.legend()
```

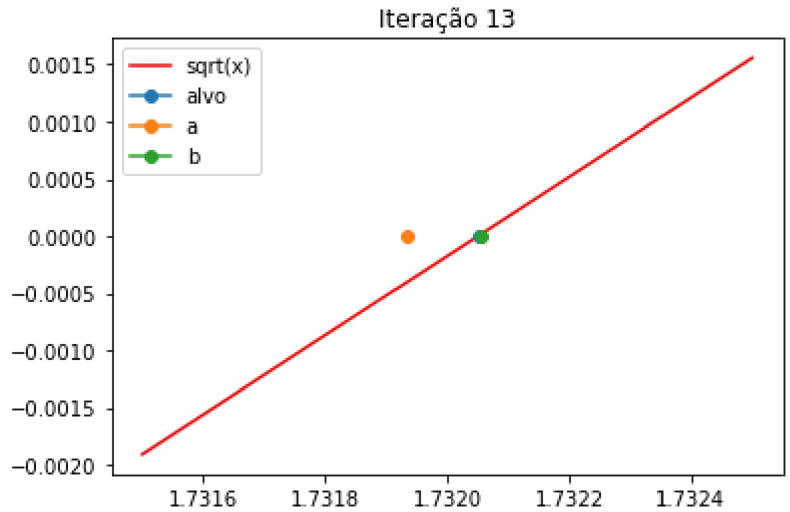
Out[8]: <matplotlib.legend.Legend at 0x2002928ef10>



In [9]:

```
xx = np.linspace(1.7315,1.7325,1000)
plt.figure(1)
y= xx**2 -3
plt.plot(xx,y,'-r',label = 'sqrt(x)')
plt.plot(1.7320508075688772,0,'-',label = 'alvo',marker = 'o')
plt.plot(1.731933,0,'-',label = 'a',marker = 'o')
plt.plot(1.732055,0,'-',label = 'b',marker = 'o')
plt.title("Iteração 13")
plt.legend()
```

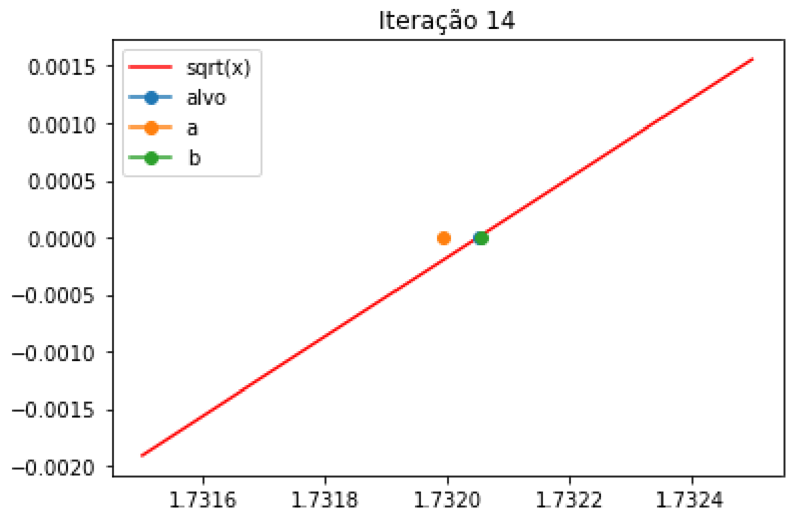
Out[9]: <matplotlib.legend.Legend at 0x2002926e6d0>



In [10]:

```
xx = np.linspace(1.7315,1.7325,1000)
plt.figure(1)
y= xx**2 -3
plt.plot(xx,y,'-r',label = 'sqrt(x)')
plt.plot(1.7320508075688772,0,'-',label = 'alvo',marker = 'o')
plt.plot(1.731994,0,'-',label = 'a',marker = 'o')
plt.plot(1.732055,0,'-',label = 'b',marker = 'o')
plt.title("Iteração 14")
plt.legend()
```

Out[10]: <matplotlib.legend.Legend at 0x2002935e220>



Com base na análise gráfica podemos constatar que na iteração 13 o ponto já havia converg
aceitável pelo exercício e quando realizado a iteração 14 como o ponto a) está muito distan

In [ ]: