```python
# TE n° 751
# sv - 28 octobre 2020
# problème 1

# On ne peut pas utiliser la méthode de bissection pour trouver
# la racine près de -1 car il n ' y a pas d'intervalle avec changement
# de signe tout près de cette racine.
# On peut cependant l'utiliser pour l'autre racine car la fonction possède
# un changement de signe dans l'intervalle [2 ; 2.5].

import numpy as np
import matplotlib.pyplot as plt
import math

# dichotomie 2
# variables f, a, b et tol la tolérance souhaitée
def dicho(f, a, b, tol):
    while b - a > tol:
        c = (a + b)/2
        if f(a)*f(c) <= 0:
            b = c
        else:
            a = c
    return a, b

def f(x):
    return x/2 - np.sin(x) + np.pi/6 - math.sqrt(3)/2

T = dicho(lambda x: x/2 - np.sin(x) + np.pi/6 - math.sqrt(3)/2, 1, 3, 10**(-10))

print('r=',T[0],'f(r)=', f(T[0]))


x = np.linspace(-2,3.50,100)
y = x/2 - np.sin(x) +np.pi/6 - np.sqrt(3)/2
plt.plot(x,y)
plt.grid(True)
plt.xlabel('x')
plt.ylabel('f(x)')
plt.show()

# r= 2.246005589258857 f(r)= -4.400912967383874e-11
```

```python
# TE n° 751
# 28 octobre 2020
# problème 2

import numpy as np
import matplotlib.pyplot as plt
import math

# variables x, x_, f et tol la tolérance souhaitée
def secante(x, x_, f, tol):
    n = 1
    while abs(f(x_) - f(x)) > tol:
        x, x_ = x_, x_ - (x_ - x) * f(x_) / (f(x_) - f(x))
        n += 1
    return x, n

# la fonction
def f(x):
    return np.e**x-3*x*x

T = secante(3, 4, f, 1e-4)
print('r_3=',T[0],'f(r_3)=', f(T[0]))

x = np.linspace(-1,5,100)
y = f(x)
plt.plot(x,y)
plt.grid(True)
plt.xlabel('x')
plt.ylabel('f(x)')
plt.show()

# r_3= 3.7330790326819776 f(r_3)= 7.859086537109761e-08
```

```
# TE n° 751
# 28 octobre 2020
# problème 3

import numpy as np
import matplotlib.pyplot as plt
import math

# variables x, x_, f et tol la tolérance souhaitée
def secante(x, x_, f, tol):
    n = 1
    while abs(f(x_) - f(x)) > tol:
        x, x_ = x_, x_ - (x_ - x) * f(x_) / (f(x_) - f(x))
        n += 1
    return x, n

# la fonction
def f(x):
    return 144*x**3-342*x**2-25*x+300

T = secante(-2, 0, f, 1e-4)
print('r_3=',T[0],'f(r_3)=', f(T[0]))

x = np.linspace(-2,0,100)
y = f(x)
plt.plot(x,y)
plt.grid(True)
plt.xlabel('x')
plt.ylabel('f(x)')
plt.show()


# r_3= -0.8333333426320924 f(r_3)= -7.85745140774452e-06
```

$$x\_ = x\_ - \frac{x\_ - x}{f(x\_) - f(x)} * f(x\_)$$

Problème 2 (5 points)

On chercher à résoudre l'équation

$$e^x - 3\,x^2 = 0$$

qui possède les deux racines $r_1 = -0.4589623$ et $r_2 = 0.91$ ainsi qu'une troisième racine r_3 située près de 4.

Utiliser la méthode de la sécante pour déterminer la troisième racine r_3 avec 4 chiffres significatifs. Afficher r_3 et $f(r_3)$ dans l'interface de l'interpréteur.

Problème 3 (7 points)

Soit la fonction

$$h(x) = 144\,x^3 - 342\,x^2 - 25\,x + 300$$

a) Montrer que la fonction h admet trois zéros distincts. 3

b) Déterminer, en utilisant la méthode de la sécante, le zéro négatif r de h avec 6 chiffres significatifs. Afficher r et $f(r)$ dans l'interface de l'interpréteur. 4

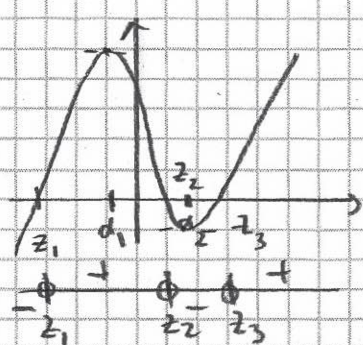
a) $h'(x) = 432x^2 - 684x - 25$

$\Delta = 511'056 > 0$ donc h' s'annule 2x en α_1 et α_2

$\alpha_1 \cong -0{,}04$ et $\alpha_2 = 1{,}62$ $h(\alpha_1) \cong 300$, $h(\alpha_2) \cong -26$

$+\ \alpha_1\ -\ \alpha_2\ +\ h'(x)$

max ↗ ↘ min ↗



b) $(3x-4)(6x+5)(8x-15) = 0$

$\Downarrow$

$-\frac{5}{6} \cong -0{,}8\bar{3}$

```
# TE n° 751
# 28 octobre 2020
# Iseni
# Aladin
# problème 5
import numpy as np
import matplotlib.pyplot as plt

def newton(x, f, df, n):
    for k in range(n):
        x = x-f(x)/df(x)
        print(x)

print(newton(1, lambda x: -5*x**3+
             7*x**2+3*x-3, lambda x: -15*x**2+14*x+3, 10))

x=np.linspace(-1,2,1000)
y=-5*x**3+7*x**2+3*x-3
plt.plot(x,y)
plt.grid()
plt.show()
```