

TP7.py

```
01 | #Exercice n°1
02 |
03 | def dichotomie(f,a,b,p):
04 |     while b-a>p:
05 |         m=(a+b)/2
06 |         if f(a)*f(m)<=0:#le < doit être remplacé par un <=
07 |             b=m
08 |         else:
09 |             a=m
10 |     return(a+b)/2
11 |
12 | #Exercice n°2
13 | from random import randint
14 |
15 | def jeu(n):
16 |     nombre=randint(1,n)
17 |     choix=-1
18 |     while choix!=nombre:
19 |         choix=int(input("Entrer un nombre : "))
20 |         if choix<nombre:
21 |             print("Trop petit")
22 |         elif choix>nombre:
23 |             print("Trop grand")
24 |         else:
25 |             print("Gagné ! ")
26 |
27 | #avec un compteur
28 |
29 | def jeu_compteur(n):
30 |     nombre=randint(1,n)
31 |     cpt=0
32 |     choix=-1
33 |     while choix!=nombre:
34 |         choix=int(input("Entrer un nombre : "))
35 |         cpt+=1
36 |         if choix<nombre:
37 |             print("Trop petit")
38 |         elif choix>nombre:
39 |             print("Trop grand")
40 |         else:
41 |             print("Gagné en ",cpt," coups ! ")
42 |
43 | #Exercice n°3
44 |
45 | def insertion(liste,x):
46 |     liste.append(x)
47 |     n=len(liste)
48 |     g,d=0,n-1
49 |     while g<=d:
50 |         m=(g+d)//2
51 |         if x<liste[m]:
52 |             d=m-1
53 |         else:
54 |             g=m+1
55 |     for j in range(g+1,n):
56 |         liste[n+g-j]=liste[n+g-j-1]
57 |     if g<n:
58 |         liste[g]=x
59 |
60 | #Exercice n°4
61 | from time import time
62 |
63 | def heron(a,epsilon):
64 |     x=1
65 |     while abs(x**2-a)>epsilon:
66 |         x=(x+a/x)/2
67 |     return(x)
68 |
69 | def solution(f,a,b,eps):
70 |     if f(a)*f(b)>0:
71 |         print("Cette fonction ne s'annule pas sur cet intervalle.")
72 |     else :
73 |         g=a
74 |         d=b
75 |         while d-g>eps:
76 |             m=(g+d)/2
77 |             if f(m)==0:
78 |                 return(m)
79 |             else: # on définit les bornes du nouvel intervalle
80 |                 if f(g)*f(m)>=0: # la solution se situe à droite de m
81 |                     g=m
82 |                 else: # la solution se situe à gauche de m
83 |                     d=m
84 |     return(m)
```

```
85|
86| def f(x):
87|     return(x**2-3)
88|
89| top=time()
90| res1=heron(3,1e-5)
91| print("Méthode du Héron : ",res1)
92| print(time()-top)
93|
94| top=time()
95| res2=solution(f,1,2,1e-5)
96| print("Méthode dichotomique :",res2)
97| print(time()-top)
```