

A Recherche d'une racine

B Recherche dans une liste triée

1 Exercices d'application

1. Ecrire une fonction `racine(f, a, b)` retournant la racine $x_0 \in [a, b]$ tel que $|f(x_0)| < 10^{-8}$.

Solution :

```

1 def racine(f, a, b):
2     assert f(a) * f(b) < 0
3
4     while abs(f((a+b)/2)) < 1e-8:
5         m = (a + b) / 2
6         if f(a) * f(m) < 0: # racine entre a et m
7             b = m
8         else: # racine entre m et b
9             a = m
10    return (a + b) / 2

```

2. Ecrire le script permettant d'obtenir une approximation de π en utilisant la fonction `cos`. Il faut définir une fonction `f` adaptée au préalable.

Solution :

```

1 def f(x):
2     return cos(x) + 1 # vaut 0 pour x = pi
3
4 val = racine(f, 3, 4) # calcul une approximation de pi à 10**-8 près

```

3. Ecrire une fonction `recherche(L, x)` retournant l'indice de l'élément `x` dans la liste `L`. On suppose ici que cet élément existe. Les éléments de `L` sont croissants.

Solution :

```

1 def recherche(L, x):
2     iG, iD = 0, len(L)
3     while iD - iG > 1: # tant que la longueur de l'intervalle est supérieur à 1
4         iM = int((iD + iG) / 2) # indice milieu
5         if x < L[iM]: # élément entre les indices iG et iM
6             iD = iM
7         else :
8             iG = iM
9     return iG

```