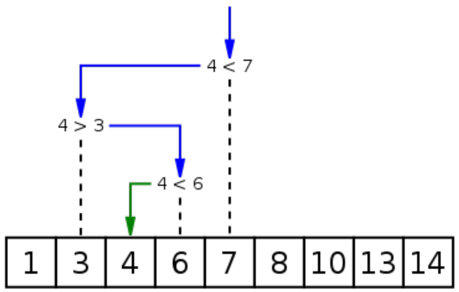
Recherche dichotomique

1 - Dans un tableau ordonné

a) Présentation

Figure 1 – Sun Zi, le génial stratège.

Dans cette partie, nous allons illustrer la méthode de dichotomie sur l'exemple de la recherche d'un élément dans un tableau trié. Nous verrons qu'il s'agit d'une méthode beaucoup plus efficace que le parcours complet du tableau. Elle fait partie des méthodes algorithmiques dites : "Diviser pour régner". *Le commandement du grand nombre est le même pour le petit nombre, ce n'est qu'une question de division en groupes.* d'après L'art de la guerre de Sun Zi (VIe siècle avant JC). Le principe consiste à déplacer deux curseurs sur une liste triée pour déterminer la position d'un élément voulu.



.

Dans l'exemple proposé, la recherche de l'élément 4 se fait en évaluant d'abord la valeur centrale de la collection (7), le terme recherché étant inférieur, on cherche au milieu de la demi-liste de gauche (entre 1 et 7). On tombe sur la valeur 3, il convient donc de rechercher entre 3 et 7, on trouve ainsi la valeur 4.

b) Implémentation

Ex. 1 :

Q 1 - Générer une liste `L` de 30 entiers croissants entre 21 et 50.

Q 2 - Proposer une fonction de recherche dichotomique `dicho(x,L)` renvoyant l'indice d'un élément x dans L. On traduira le principe suivant :

- on initialise deux curseurs `a` et `b` à chaque extrémité de la liste.
- tant que l'écart entre `a` et `b` est strictement supérieur à 1,
  - on calcule l'indice `m` de l'élément au milieu de la liste
  - si l'élément voulu est supérieur à `L[m]` , on change `a` en `m`
  - sinon on change `b` en `m`
- retourner la valeur `a` ou `b` par comparaison de `L[a]` et `L[b]` avec x.

Q 3 - Vérifier que votre code fonctionne pour la recherche de 22. On attend un affichage du type :

```
Position de 21 dans L : 0
Position de 22 dans L : 1
```

c) Calcul du coût

Ex. 2 :

Soit la liste suivante `L = [11, 13, 37, 55, 56, 57, 66, 72, 80, 81]` composée de 10 éléments.

Q 1 - Combien d'opérations sont nécessaires à la fonction suivante pour trouver la valeur 81 :

```python
def trouve(x,L):
    for k in range(len(L)):
        if x == L[k] : return k
```

Q 2 - Modifier le code de la fonction `dicho` pour renvoyer le nombre d'opérations permettant d'obtenir la même valeur par dichotomie.

Q 3 - Montrer que pour une liste de n éléments, le nombre d'itérations est de l'ordre de $\lfloor \log_2 n \rfloor = \lfloor \frac{\ln n}{\ln 2} \rfloor$.

2 - Recherche du 0 d'une fonction

Un problème stationnaire est le recherche d'une solution x d'un problème écrit sous la forme

$$f(x) = C$$

où f est une fonction et C une constante. Le théorème de Bolzano cas particulier du théorème des valeurs intérmédiaires assure l'existence d'une solution à l'équation $f(x) = 0$.

On considère deux nombres a et b et une fonction f continue sur l'intervalle $[a, b]$ telle que $f(a)$ et $f(b)$ soient de signes opposés. Supposons que nous voulions résoudre l'équation $f(x) = 0$. Nous savons d'après le théorème des valeurs intermédiaires que f doit avoir au moins un zéro dans l'intervalle $[a, b]$. La méthode de dichotomie consiste à diviser l'intervalle en deux en calculant $c = (a + b)/2$. Il y a maintenant deux possibilités :

- soit $f(a)$ et $f(c)$ sont de signes contraires,
- soit $f(c)$ et $f(b)$ sont de signes contraires.

L'algorithme de dichotomie est alors appliqué au sous-intervalle dans lequel le changement de signe se produit

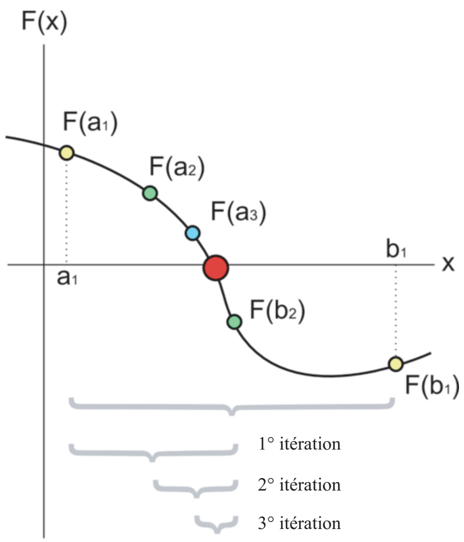
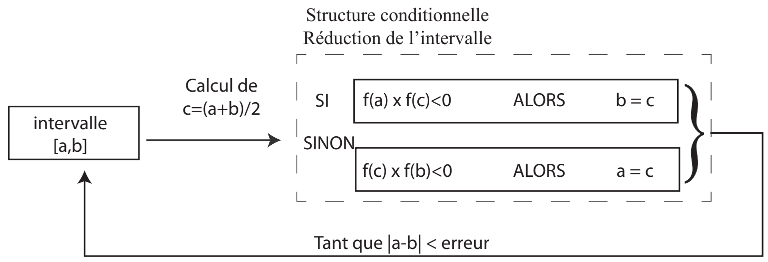


Figure 2 – Schéma de l'algorithme de dichotomie

Le schéma de résolution est le suivant :



Remarque 1 :
Dans le cas où la fonction admet plusieurs 0, l'algorithme de dichotomie détermine l'un d'entre eux.

Ex. 3 :

Q 1 - Réaliser une fonction `zero` de paramètre `f, a, b` et `e` qui renvoie une valeur approchée d'un zéro de la fonction f à e près. On testera la fonction sur une fonction f de la forme $f(x) = x^2 - 2$ avec $a = 0$ et $b = 2$.

```
valeur approchee de 2**0.5 : 1.4142
```

Q 2 - Justifier le résultat obtenu avec $a = -2$ et $b = 0$ ainsi que $a = -2$ et $a = 2$

Q 3 - Montrer que le nombre d'itérations est de l'ordre de $\lfloor \log_2 \frac{b-a}{e} \rfloor = \lfloor \frac{\ln \frac{b-a}{e}}{\ln 2} \rfloor$.

3 - Exponentiation rapide

Pour calculer la puissance d'un nombre, la méthode brutale consiste à effectuer n multiplication :

$$x^n = \underbrace{x \times x \times \cdots \times x}_{n fois}$$

Il est cependant possible de réduire le nombre d'opérations en exploitant les résultats suivants :

- si n est pair alors : $x^n = \left(x^2\right)^{n//2}$
- si n est impair alors : $x^n = x \times x^{n-1} = x \times \left(x^2\right)^{n//2}$

Ex. 4 :

Q 1 - Proposer une fonction `puissance(x,n)` qui exploite la propriété ci-dessus.

Q 2 - Montrer que, si $n = 2^p$, le nombre d'opérations est p.