

TP Informatique 11

 On rappelle qu'un script (fichier *.py) doit être enregistré et exécuté (touche F5) pour que les fonctions saisies dans le script soient utilisables dans la console et que la combinaison de touches **Ctrl+C** permet de casser une boucle infinie.

Exercice 1

On rappelle le principe d'une recherche dichotomique d'un nombre `elt` dans une liste de nombres triée non vide `L` :

- on considère l'élément au milieu de `L` ;
- si c'est `elt`, on s'arrête ;
- si `elt` est plus petit que l'élément du milieu, on se place sur la moitié de gauche, sinon on se place sur la moitié de droite ;
- on poursuit ce processus tant qu'on n'a pas rencontré `elt` et que la zone de recherche n'est pas vide.

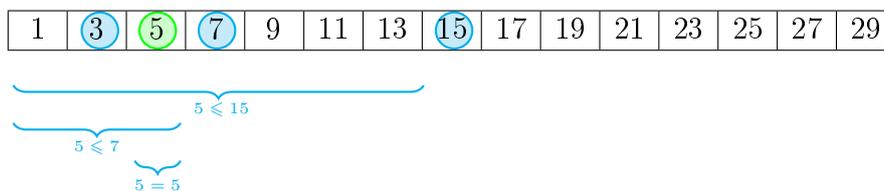


FIGURE 1 – Recherche dichotomique de 5 dans $[1, 3, \dots, 29]$

Écrire une fonction `rech1(elt,L)` d'arguments `elt` un nombre et `L` une liste de nombres triée non vide qui effectue une recherche dichotomique de `elt` dans `L` et renvoie `True,ind` si `elt` est présent dans `L` avec `ind` un indice de `elt` dans `L` et `False,ind` sinon (`ind` étant un indice quelconque dans ce cas).

Tester la fonction `rech1` avec la liste $[1, 3, 5, \dots, 17, 19]$ et les entiers 1, 19, 11, 12, 0, 20.

Exercice 2

Pour x, y réels, on définit la distance entre x et y par $d(x, y) = |x - y|$. On se propose de programmer de deux façons différentes le calcul de la distance minimale entre deux éléments d'indices distincts d'une liste `L` de nombres :

- si la liste est préalablement triée, la distance minimale est réalisée entre deux valeurs consécutives ;
- si la liste n'est pas triée, il faut examiner toutes les distances possibles entre deux valeurs d'indices distincts.

Par exemple, avec la liste $[1, 7, 4, 3]$:

- pour la première méthode, on effectue le tri en $[1, 3, 4, 7]$, on compare $3 - 1$, $4 - 3$, $7 - 4$ et on renvoie $4 - 3$;
- pour la deuxième méthode, on compare $|3 - 1|$, $|4 - 1|$, $|4 - 3|$, $|7 - 4|$, $|7 - 3|$, $|7 - 1|$ et on renvoie $|4 - 3|$.

1. Tester l'instruction `sorted` sur différentes listes de nombres. Que réalise-t-elle ?
2. Écrire une fonction `dmin_sort(L)` d'argument `L` une liste de nombres et qui renvoie la distance minimale entre deux éléments d'indices distincts de `L`. On pourra utiliser `sorted` et on effectuera une seule boucle `for`.
3. Écrire une fonction `dmin(L)` d'argument `L` une liste de nombres et qui renvoie la distance minimale entre deux éléments d'indices distincts de `L`. On n'utilisera pas de tri sur cette liste.
4. L'instruction qui suit permet de générer une liste `L` de 10000 nombres tirés au hasard dans `[0; 1]` :

```
import numpy.random as rd
L=rd.random(10000)
```

Tester les fonctions `dmin` et `dmin_sort` sur cette liste `L`. Que constate-t-on ?

Exercice 3

1. Dans la console, saisir successivement les instructions suivantes :

```
a=0, b=0, id(a), id(b), a=1000, b=1000, id(a), id(b)
```

Qu'observe-t-on ?

2. Dans la console, saisir les instructions suivantes :

```
c=1000, a=c, b=c, id(a), id(b)
```

Constata-t-on le même phénomène que précédemment ?

3. Écrire un programme, qui détermine, sans recherche dichotomique, le seuil de changement du phénomène observé à la première question. On précise qu'il est possible de casser une boucle `for` avec l'instruction `break`.
4. Écrire un programme qui détermine, de manière dichotomique, le seuil de changement du phénomène observé à la première question.