

## Feuille de TD n° 4 : complexité

Contenu : rappels sur les équivalents et relations de comparaison, complexité d'algorithmes.

**Exercice 1.** Les deux assertions  $u_n = v_n + o(1)$  et  $u_n \sim v_n$  sont-elles comparables ?

**Exercice 2.** Trouver un exemple de suite réelle  $(u_n)$  telle que  $u_n = o(e^{\alpha n})$  pour tout  $\alpha > 0$  et  $n^k = o(u_n)$  pour tout  $k > 0$ .

**Exercice 3.** Donner un équivalent de  $2\sqrt{n} - \sqrt{n+1} - \sqrt{n-1}$ , puis de  ${}^{n+1}\sqrt{n+1} - \sqrt[n]{n}$ .

**Exercice 4.** Déterminer les limites des suites suivantes :

$$\left(1 + \sin \frac{1}{n}\right)^n, \quad \left(n \sin \frac{1}{n}\right)^{n^2}, \quad (3\sqrt[n]{2} - 2\sqrt[n]{3})^n.$$

**Exercice 5.** Quelle est la complexité des fonctions suivantes ? Autant que possible, dire ce que retourne chaque fonction.

```
1 def f1(n):
2     x = 0
3     for i in range(n):
4         for j in range(n):
5             x += 1
6     return x
```

```
1 def f4(n):
2     x, i = 0, n
3     while i > 1:
4         x += 1
5         i //= 2
6     return x
```

```
1 def f2(n):
2     x = 0
3     for i in range(n):
4         for j in range(i):
5             x += 1
6     return x
```

```
1 def f5(n):
2     x, i = 0, n
3     while i > 1:
4         for j in range(n):
5             x += 1
6             i //= 2
7     return x
```

```
1 def f3(n):
2     x = 0
3     for i in range(n):
4         j = 0
5         while j * j < i:
6             x += 1
7             j += 1
8     return x
```

```
1 def f6(n):
2     x, i = 0, n
3     while i > 1:
4         for j in range(i):
5             x += 1
6             i //= 2
7     return x
```

**Exercice 6.** Quelle est la complexité des cinq algorithmes de tri vus en cours ?

**Exercice 7.** L'algorithme suivant prend en entrée une liste de nombres déjà triée  $L$  et un nombre  $x$ . Que fait-il ? Quelle est sa complexité ?

```
1 def f(x, L):
2     i, j = 0, len(L)
3     while i < j:
4         k = (i + j) // 2
5         if L[k] == x:
6             return True
7         elif L[k] > x:
8             j = k
9         else:
10            i = k + 1
11    return False
```