

# Les tableaux

---

Pour lire le contenu d'un tableau...

il faut une **boucle pour aller lire chaque case** !

Si le tableau a été prévu trop court au début, **impossible de changer sa longueur...** il faut une boucle pour le recopier dans un tableau plus grand !

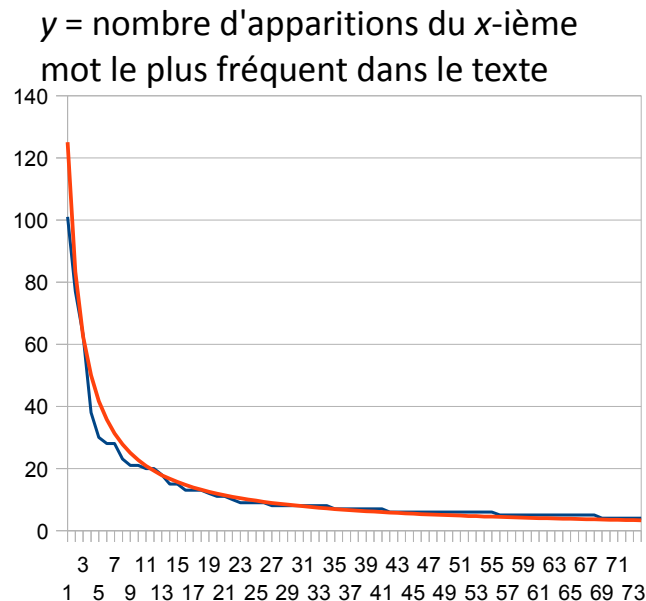
Possibilité de créer des **tableaux de tableaux...**



# Le nombre d'apparitions d'un mot dans un texte

## La "minute mathématique"

La loi de Zipf prédit la courbe du nombre d'apparitions des mots les plus fréquents d'un texte.



$$y = 250 / (x + 1)$$

x = numéro du mot (1 pour le plus fréquent, 2 pour le 2° plus fréquent...)

Fonctionne pour n'importe quel texte assez long...

# La boucle “for” / “Pour tout...”

La boucle “for” / “Pour tout”

Une boucle pour **parcourir tous les entiers entre deux valeurs entières.**

Algorithme **DessineHistogramme**

**Entrée :** tableau de chaînes de caractères *Mots* et tableau d'entiers *NbApparitions*.

**Variable :** entier *compteur*

Début

*compteur* ← 1

**Tant que** *compteur* < **Longueur**(*Mots*)+1 **faire :**

**dessineRectanglePlein**(*compteur*\*4-4,  
        50-10\***Case**(*NbApparitions*,*compteur*),  
        4,10\***Case**(*NbApparitions*,*compteur*),  
        **couleurRGB**(0,0,255))

*compteur* ← 1 + *compteur*

**Fin TantQue**

Fin

En Java :

```
int compteur;  
compteur=1;  
while (compteur<mots.length+1) {  
    ...  
}
```

Algorithme **DessineHistogramme**

**Entrée :** tableau de chaînes de caractères *Mots* et tableau d'entiers *NbApparitions*.

**Variable :** entier *compteur*

Début

**Pour** *compteur* de 1 à **Longueur**(*Mots*) **faire :**

**dessineRectanglePlein**(*compteur*\*4-4,  
        50-10\***Case**(*NbApparitions*,*compteur*),  
        4,10\***Case**(*NbApparitions*,*compteur*),  
        **couleurRGB**(0,0,255))

*compteur* ← 1 + *compteur*

**Fin Pour**

Fin

*déclaration +  
initialisation*      *condition d'arrêt*      *mise à jour*

```
for (int compteur=1; compteur<mots.length+1; compteur++) {  
    ...  
}
```