

DUT MMI – IUT de Marne-la-Vallée
06/11/2018
M1202 - Algorithmique

Cours 4
Tableaux et boucles
(suite + méthodologie)

Sources

- *Le livre de Java premier langage*, d'A. Tasso
- Cours INF120 de J.-G. Luque
- Cours FLIN102 de l'Université Montpellier 2
- Cours de J. Henriet : <http://julienhenriet.olymp-network.com/Algo.html>
- <http://xkcd.com>, <http://xkcd.free.fr>

Plan des cours 3 & 4 – Tableaux et boucles

- Résumé des épisodes précédents
- Les tableaux
- Affichage du contenu d'un tableau
- La boucle “for” / “pour tout”
- Dessin d'un graphique à partir du contenu d'un tableau

Résumé de l'épisode précédent

Tableaux :

- pour stocker un ensemble de valeurs **de même type** (de préférence)
- **une** valeur par case
- **nombre de cases fixé** à l'initialisation du tableau (de préférence)
- boucle pour **parcourir le tableau**

Boucles :

- attention à l'**initialisation** et à la **condition d'arrêt** (premier et dernier passage dans la boucle)
- boucle **Tant que** et boucle **Pour tout** (en TD/TP)

Plan des cours 3 & 4 – Tableaux et boucles

- Résumé des épisodes précédents
- Les tableaux
- Affichage du contenu d'un tableau
- La boucle “for” / “pour tout”
- Dessin d'un graphique à partir du contenu d'un tableau

La boucle “for” / “Pour tout...”

Pour parcourir tous les entiers entre deux valeurs entières.

En pseudo-code :

Pour tout entier i de 1 à 42 faire :

...

Fin Pour

En Javascript :

```
for (var i=1; i<43; i++) {
```

```
    ...
```

```
}
```

La boucle “for” / “Pour tout...”

Exemple : **parcours des cases d'un tableau**

En pseudo-code avec Tant que :

Variables : tableau d'entiers *tab*, entier *i*

$i \leftarrow 1$

Tant que $i < \text{longueur}(tab)+1$ faire :

[des choses avec la *i*-ième case du tableau **case**(*tab*,*i*)...]

$i \leftarrow i+1$

Fin Tant que

En pseudo-code avec Pour :

Variables : tableau d'entiers *tab*, entier *i*

Pour *i* de 1 à **longueur**(*tab*) faire :

[des choses avec la *i*-ième case du tableau **case**(*tab*,*i*)...]

Fin Pour

La boucle “for” / “Pour tout...”

Pour parcourir tous les entiers entre deux valeurs entières.

En pseudo-code :

Pour tout entier i de 1 à 42 faire :

...

Fin Pour

En Javascript :

```
for (var i=1; i<43; i++) {  
    ...  
}
```

En Javascript avec while :

```
var i=1;  
while (i<43) {  
    ...  
    i++;  
}
```


La boucle “for” / “Pour tout...”

Pour parcourir tous les entiers entre deux valeurs entières.

En pseudo-code :

Pour tout entier i de 1 à 42 faire :

...

Fin Pour

En Javascript :

```
for(var i=1, i<43, i++) {  
  ...  
}
```

*déclaration +
initialisation*

En Javascript avec while :

```
var i=1;  
while (i<43) {  
  ...  
  i++;  
}
```

La copie d'un tableau

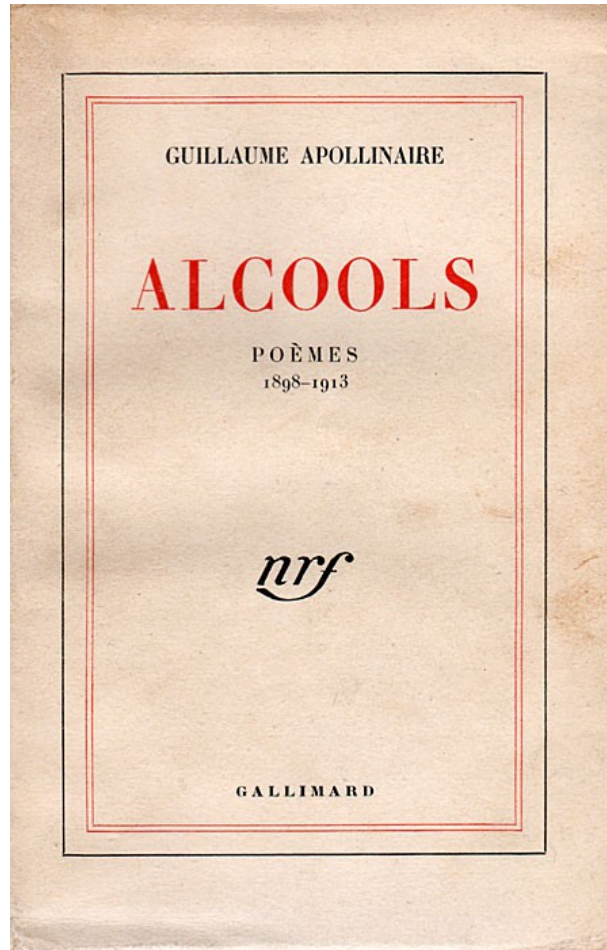
Pour copier le contenu d'un tableau d'entiers *t1* dans un nouveau tableau d'entiers *t2*.

En Javascript :

Plan des cours 3 & 4 – Tableaux et boucles

- Résumé des épisodes précédents
- Les tableaux
- Affichage du contenu d'un tableau
- La boucle “for” / “pour tout”
- Dessin d'un graphique à partir du contenu d'un tableau

Graphique du nombre d'apparitions des mots d'un texte



J'ai cueilli ce brin de bruyère
L'automne est morte souviens-t'en
Nous ne nous verrons plus sur terre
Odeur du temps brin de bruyère
Et souviens-toi que je t'attends

Graphique du nombre d'apparitions des mots d'un texte

Un tableau
de chaînes de
caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

J'ai cueilli ce brin de bruyère
L'automne est morte souviens-t'en
Nous ne nous verrons plus sur terre
Odeur du temps brin de bruyère
Et souviens-toi que je t'attends

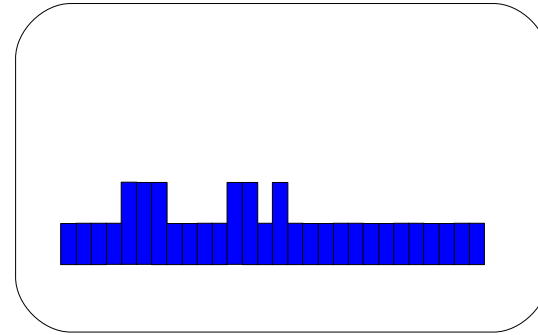
Un tableau d'entiers *nbApparitions*

Graphique du nombre d'apparitions des mots d'un texte

Un tableau
de chaînes de
caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

Résultat voulu :



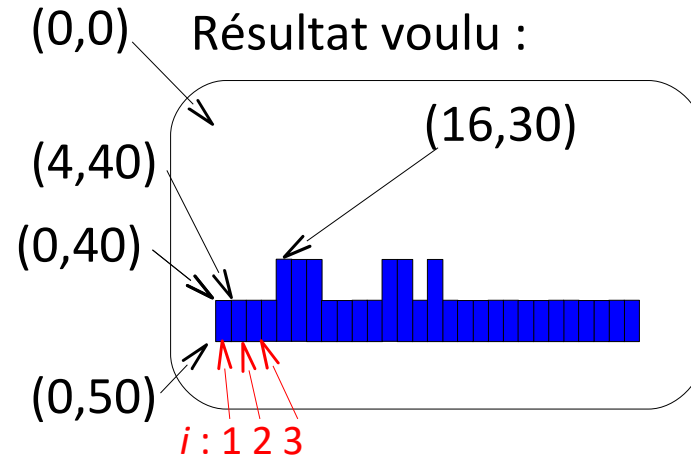
Un tableau d'entiers *nbApparitions*

Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que ... faire :

Fin Tant que

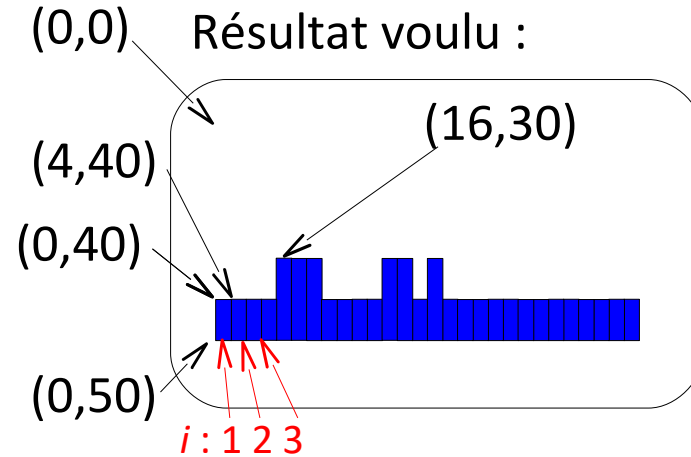
Fin

Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

Fin Tant que

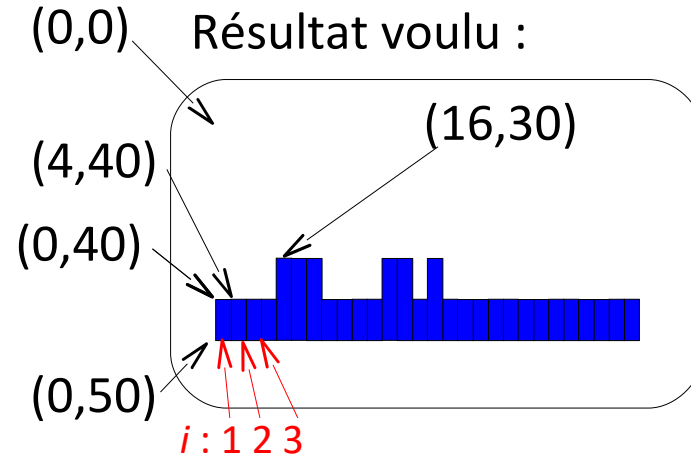
Fin

Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

// dessin du 1er et du 2e rectangle :

dessineRectanglePlein(*imageDeFond*, 0, 40, 4, 10, "blue")

dessineRectanglePlein(*imageDeFond*, 4, 40, 4, 10, "blue")

Fin Tant que

Fin

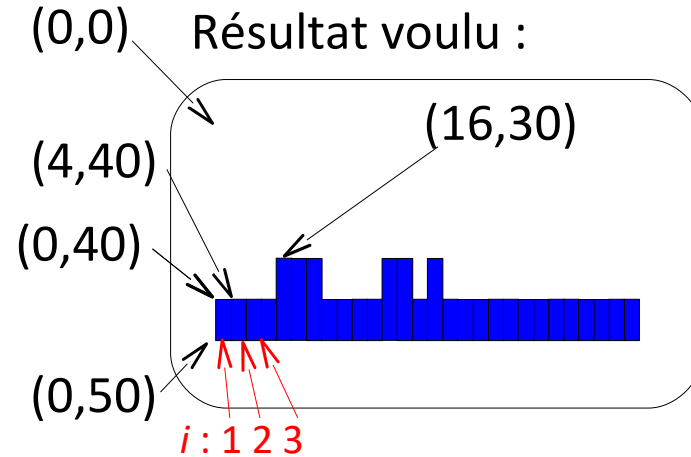
Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

<i>i</i>	<i>x</i>	$4i$
1	0	4
2	4	8
3	8	12
4	12	16
...		

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

// dessin du 1er et du 2e rectangle :

dessineRectanglePlein(*imageDeFond*, 0, 40, 4, 10, "blue")

dessineRectanglePlein(*imageDeFond*, 4, 40, 4, 10, "blue")

Fin Tant que

Fin

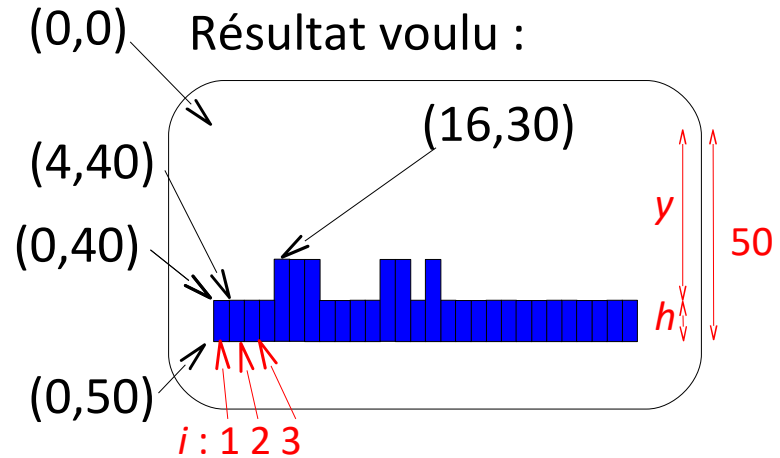
Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

<i>i</i>	<i>x</i>	$4i$
1	0	4
2	4	8
3	8	12
4	12	16
...		

Un tableau d'entiers *nbApparitions*



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

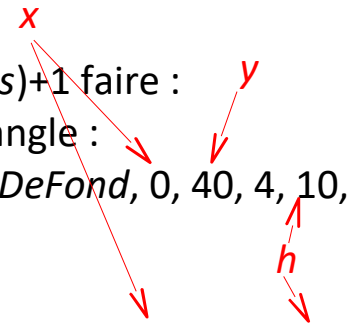
// dessin du 1er et du 2e rectangle :

dessineRectanglePlein(*imageDeFond*, 0, 40, 4, 10, "blue")

dessineRectanglePlein(*imageDeFond*, 4, 40, 4, 10, "blue")

Fin Tant que

Fin



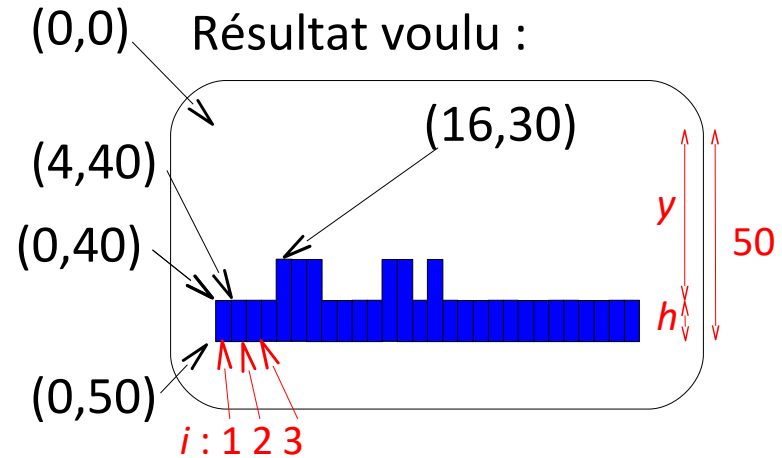
Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

<i>i</i>	<i>x</i>	$4i$
1	0	4
2	4	8
3	8	12
4	12	16
...		

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

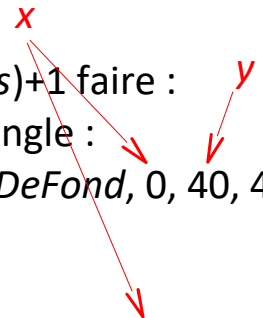
// dessin du 1er et du 2e rectangle :

dessineRectanglePlein(*imageDeFond*, 0, 40, 4, 10, "blue")

dessineRectanglePlein(*imageDeFond*, 4, 40, 4, 10, "blue")

Fin Tant que

Fin



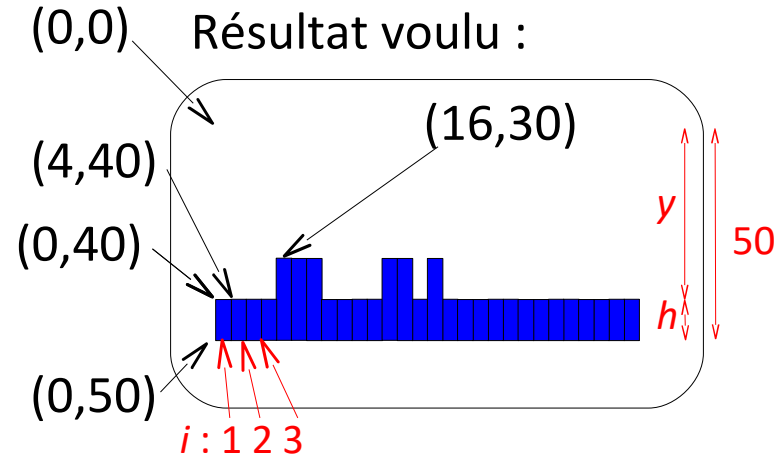
Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

<i>i</i>	<i>x</i>	$4i$
1	0	4
2	4	8
3	8	12
4	12	16
...		

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

// dessin du 1er et du 2e rectangle :

dessineRectanglePlein(*imageDeFond*, 0, 40, 4, 10, "blue")

dessineRectanglePlein(*imageDeFond*, 4, 40, 4, 10, "blue")

Fin Tant que

Fin

$x = 4(i-1)$

y

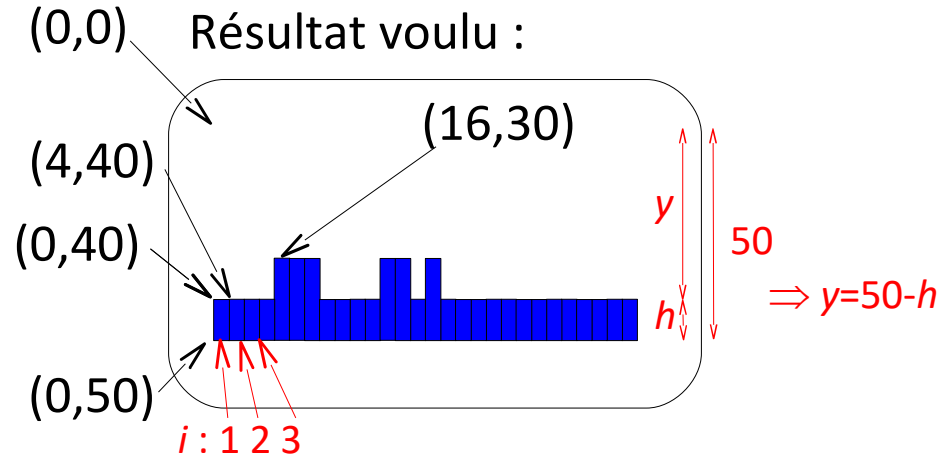
$h = 10 * \text{case}(\text{nbApparitions}, i)$

Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

// dessin du 1er et du 2e rectangle :

dessineRectanglePlein(*imageDeFond*, 0, 40, 4, 10, "blue")

dessineRectanglePlein(*imageDeFond*, 4, 40, 4, 10, "blue")

Fin Tant que

Fin

$x = 4(i-1)$

$y = 50 - 10 * \text{case}(\text{nbApparitions}, i)$

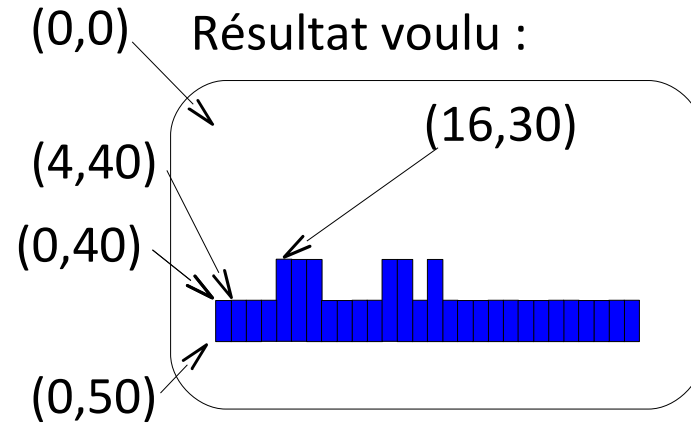
$h = 10 * \text{case}(\text{nbApparitions}, i)$

Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

// dessin des rectangles :

dessineRectanglePlein(*imageDeFond*, $4 * (i - 1)$,
 $50 - 10 * \text{case}(\text{nbApparitions}, i)$,
 4 , $10 * \text{case}(\text{nbApparitions}, i)$,
couleur)

Fin Tant que

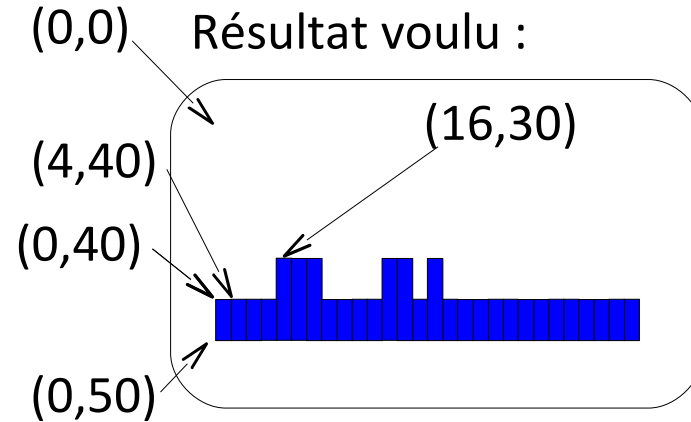
Fin

Graphique du nombre d'apparitions des mots d'un texte

Un tableau de chaînes de caractères *mots*

j	1
ai	1
cueilli	1
ce	1
brin	2
de	2
bruyère	2
l	1
automne	1
est	1
morte	1
souviens	2
t	2
en	1
nous	2
ne	1
verrons	1
plus	1
sur	1
terre	1
odeur	1
du	1
temps	1
et	1
toi	1
que	1
je	1
attends	1

Un tableau d'entiers *nbApparitions* ↗



Algorithme **dessineHistogramme**

Entrée : chaîne de caractères *imageDeFond*, entiers *abscisseBG*, *ordonneeBG* et *hauteurMin*, tableau d'entiers *nbApparitions*, chaîne de caractères *couleur*

Variable : entier *i*

Début

$i \leftarrow 1$

Tant que $i < \text{longueur}(\text{nbApparitions}) + 1$ faire :

// dessin des rectangles :

dessineRectanglePlein(*imageDeFond*, $4 * (i - 1)$,
 $50 - 10 * \text{case}(\text{nbApparitions}, i)$,
 4 , $10 * \text{case}(\text{nbApparitions}, i)$,
couleur)

$i \leftarrow i + 1$

Fin Tant que

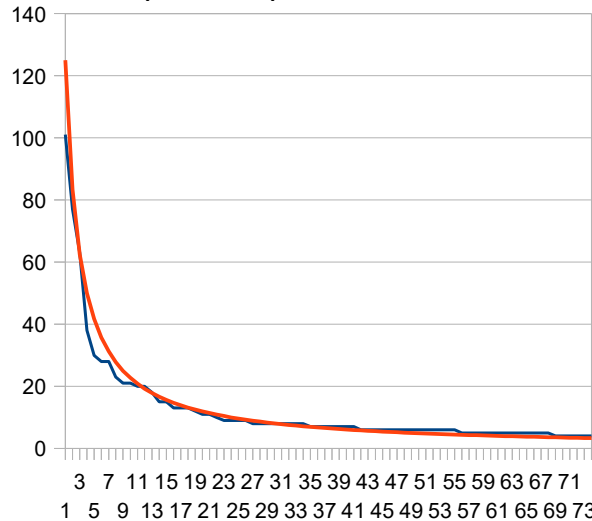
Fin

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 = nombre d'apparitions du x -ième mot le plus fréquent dans le texte



— Nombre réel d'apparition des mots
— Nombre estimé d'apparition des mots

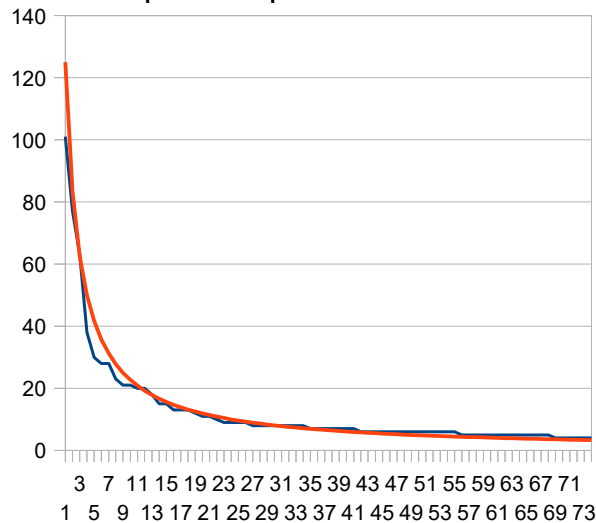
x = numéro du mot (1 pour le plus fréquent, 2 pour le 2° plus fréquent...)

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 = nombre d'apparitions du x -ième mot le plus fréquent dans le texte



— Nombre réel d'apparition des mots
— Nombre estimé d'apparition des mots

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

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

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