

Bytes et tailles

Ivan Noyer

Définition

Définition

On appelle *byte* (ou *multiplets* en français) la plus petite unité « logiquement » adressable par un programme sur un ordinateur. Aujourd'hui, le besoin d'une structure commune pour le partage des données a fait que le byte de 8 bits, ou 1 octet, s'est généralisé en informatique

(Wikipedia)

Remarque

Le langage C ne peut pas réserver une quantité de mémoire inférieure à un byte.

Bon à savoir

- Dans les années 70 et auparavant, il existait des processeurs avec des bytes de tailles très variables.

Bon à savoir

- Dans les années 70 et auparavant, il existait des processeurs avec des bytes de tailles très variables.
- Actuellement, il existe encore des processeurs avec des tailles de bytes de 4 bits voire moins (notamment pour la programmation des automates industriels).

Bon à savoir

- Dans les années 70 et auparavant, il existait des processeurs avec des bytes de tailles très variables.
- Actuellement, il existe encore des processeurs avec des tailles de bytes de 4 bits voire moins (notamment pour la programmation des automates industriels).
- Beaucoup de microprocesseurs adressent physiquement la mémoire avec des mots de plusieurs bytes afin d'augmenter les performances.

Bon à savoir

- Dans les années 70 et auparavant, il existait des processeurs avec des bytes de tailles très variables.
- Actuellement, il existe encore des processeurs avec des tailles de bytes de 4 bits voire moins (notamment pour la programmation des automates industriels).
- Beaucoup de microprocesseurs adressent physiquement la mémoire avec des mots de plusieurs bytes afin d'augmenter les performances.
- En C, l'opérateur unaire `sizeof` donne la taille en bytes de son opérande. L'opérande peut être un spécificateur de type ou une expression.

Bon à savoir

- Dans les années 70 et auparavant, il existait des processeurs avec des bytes de tailles très variables.
- Actuellement, il existe encore des processeurs avec des tailles de bytes de 4 bits voire moins (notamment pour la programmation des automates industriels).
- Beaucoup de microprocesseurs adressent physiquement la mémoire avec des mots de plusieurs bytes afin d'augmenter les performances.
- En C, l'opérateur unaire `sizeof` donne la taille en bytes de son opérande. L'opérande peut être un spécificateur de type ou une expression.
- Nous prendrons 1 octet comme taille de byte pour la suite du cours.

Utilité de `sizeof`

- Il est souvent utile de connaître la taille d'un type de donnée (en particulier d'une *structure*).

Utilité de `sizeof`

- Il est souvent utile de connaître la taille d'un type de donnée (en particulier d'une *structure*).
- En C la taille des types primitifs est une constante exprimée en byte. Cependant, la taille réelle en octet du byte dépend des machines et de l'OS. Pour assurer la portabilité du code, il est préférable de réserver une zone mémoire en byte plutôt qu'en octet.

Utilité de `sizeof`

- Il est souvent utile de connaître la taille d'un type de donnée (en particulier d'une *structure*).
- En C la taille des types primitifs est une constante exprimée en byte. Cependant, la taille réelle en octet du byte dépend des machines et de l'OS. Pour assurer la portabilité du code, il est préférable de réserver une zone mémoire en byte plutôt qu'en octet.
- Ci-dessous, la fonction `malloc` alloue une zone mémoire faisant la taille (en bytes) de 10 entiers. Elle retourne un pointeur sur cette zone :

```
1 int *pointeur ;  
2 pointeur = (int *) malloc(10 * sizeof (int));
```

Taille d'un tableau

Pour un tableau de 5 entiers `int t[5];` :

- la commande `sizeof(t)` donne la taille en bytes. Aucune inclusion n'est nécessaire pour `sizeof`.

En revanche, sa valeur de retour est un entier de type `size_t` (lequel est défini dans l'entête standard `stdio.h`).

Taille d'un tableau

Pour un tableau de 5 entiers `int t[5];` :

- la commande `sizeof(t)` donne la taille en bytes. Aucune inclusion n'est nécessaire pour `sizeof`.
En revanche, sa valeur de retour est un entier de type `size_t` (lequel est défini dans l'entête standard `stdio.h`).
- la commande `sizeof (*t)` (ou encore `sizeof *t`) donne la taille du type des éléments de `t`. On peut aussi utiliser `sizeof(int)`

Taille d'un tableau

Pour un tableau de 5 entiers `int t[5];` :

- la commande `sizeof(t)` donne la taille en bytes. Aucune inclusion n'est nécessaire pour `sizeof`.
En revanche, sa valeur de retour est un entier de type `size_t` (lequel est défini dans l'entête standard `stdio.h`).
- la commande `sizeof (*t)` (ou encore `sizeof *t`) donne la taille du type des éléments de `t`. On peut aussi utiliser `sizeof(int)`
- `sizeof(t[0])` renvoie la taille du 1er élément de `t`. C'est la taille d'un `int`.

Taille d'un tableau

Pour un tableau de 5 entiers `int t[5];` :

- la commande `sizeof(t)` donne la taille en bytes. Aucune inclusion n'est nécessaire pour `sizeof`.
En revanche, sa valeur de retour est un entier de type `size_t` (lequel est défini dans l'entête standard `stdio.h`).
- la commande `sizeof (*t)` (ou encore `sizeof *t`) donne la taille du type des éléments de `t`. On peut aussi utiliser `sizeof(int)`
- `sizeof(t[0])` renvoie la taille du 1er élément de `t`. C'est la taille d'un `int`.
- Si on ne connaît pas le nombre d'éléments de `t`, on le retrouve par `sizeof t/ sizeof *t;`. Cette technique fait bien ce qu'on veut mais uniquement dans le bloc où `t` est déclaré!

Les parenthèses ne sont pas obligatoires avec `sizeof` s'il s'agit de déterminer la taille d'une expression. Elles le deviennent pour le nom d'un type.

Exercice

Dans un fichier `taille.c` :

- Directement dans le `main`, afficher la taille des types `size_t`, `int`, `int *`.

Exercice

Dans un fichier `taille.c` :

- Directement dans le `main`, afficher la taille des types `size_t`, `int`, `int *`.
- Directement dans le `main`, créer maintenant un tableau de 5 entiers et afficher sa taille avec la technique précédente.

Exercice

Dans un fichier `taille.c` :

- Directement dans le `main`, afficher la taille des types `size_t`, `int`, `int *`.
- Directement dans le `main`, créer maintenant un tableau de 5 entiers et afficher sa taille avec la technique précédente.
- Écrire une fonction `int tailleTab (int t[])` qui prend en paramètre un tableau et affiche sa taille selon la technique précédente. Appeler maintenant dans le `main` cette fonction avec un tableau de 5 entiers et comparer avec l'affichage précédent. Expliquer.

Multiples de l'octet

- Un octet vaut 8 bits.

Multiples de l'octet

- Un octet vaut 8 bits.
- Les tailles de fichiers ou de variables sont données en multiples de l'octet. Les multiples décimaux sont les plus utilisés (kilo, méga, giga, téra) mais ne sont pas recommandés par la Commission électrotechnique internationale.

Multiples de l'octet

- Un octet vaut 8 bits.
- Les tailles de fichiers ou de variables sont données en multiples de l'octet. Les multiples décimaux sont les plus utilisés (kilo, méga, giga, téra) mais ne sont pas recommandés par la Commission électrotechnique internationale.
- Une nouvelle norme à été créée en 1988 pour noter les multiples de $2^{10} = 1024$: les kibi (kilo binaire), mébi (méga binaire), gibi (giga binaire) etc.

Multiples de l'octet

- Un octet vaut 8 bits.
- Les tailles de fichiers ou de variables sont données en multiples de l'octet. Les multiples décimaux sont les plus utilisés (kilo, méga, giga, téra) mais ne sont pas recommandés par la Commission électrotechnique internationale.
- Une nouvelle norme a été créée en 1988 pour noter les multiples de $2^{10} = 1024$: les kibi (kilo binaire), mébi (méga binaire), gibi (giga binaire) etc.
- Certains systèmes d'exploitation annoncent faussement les quantités d'octets avec un suffixe décimal. Par exemple une quantité de 32 Go peut être affichée correctement comme 29.8 Gio ou improprement comme 29.8 Go sur certains systèmes d'exploitations.

Multiples décimaux de l'octet

Multiples décimaux de l'octet dans le SI et mauvais usage :

Nom	Symbole	Valeur en octets	Mésusage
kilooctet	ko	10^3	2^{10}
mégaoctet	Mo	10^6	2^{20}
gigaoctet	Go	10^9	2^{30}
téraoctet	To	10^{12}	2^{40}
pétaoctet	Po	10^{15}	2^{50}
exaoctet	Eo	10^{18}	2^{60}
zettaoctet	Zo	10^{21}	2^{70}
yottaoctet	Yo	10^{24}	2^{80}

ko, Mo, Go (noter les majuscules et minuscules) ♥

Multiples binaires de l'octet

Nom	Symbole	Valeur en octets	
kibioctet	Kio	2^{10}	1024 octets
mébioctet	Mio	2^{20}	1024 Kio
gibioctet	Gio	2^{30}	1024 Mio
tébioctet	Tio	2^{40}	1024 Gio
pébioctet	Pio	2^{50}	1024 Tio
exbioctet	Eio	2^{60}	1024 Pio
zébioctet	Zio	2^{70}	1024 Eio
yobioctet	Yio	2^{80}	1024 Zio

Kio, Mio, Gio (noter les majuscules et minuscules) ♡

Taille d'un fichier

La commande `du`

- Dans un terminal depuis le répertoire courant

```
$ du -h
```

Donne la taille du répertoire courant et ses sous-répertoires

Taille d'un fichier

La commande `du`

- Dans un terminal depuis le répertoire courant

```
$ du -h
```

Donne la taille du répertoire courant et ses sous-répertoires

- Pour avoir la taille des fichiers du répertoire, utiliser `*`. Option `-h` pour « Human redable » : résultats en Kilo-octet, Mega-octet, Giga-octet...

```
$ du -h *
```

Taille d'un fichier

La commande `du`

- Dans un terminal depuis le répertoire courant

```
$ du -h
```

Donne la taille du répertoire courant et ses sous-répertoires

- Pour avoir la taille des fichiers du répertoire, utiliser `*`. Option `-h` pour « Human redable » : résultats en Kilo-octet, Mega-octet, Giga-octet...

```
$ du -h *
```

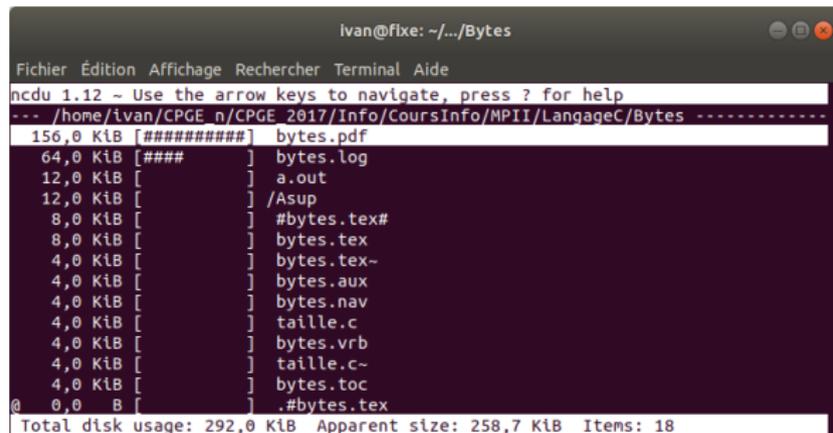
- L'option `-s` permet de n'afficher que le total de la taille du répertoire.

```
$ du -sh
```

Taille d'un fichier

La commande `ncdu`

Plus conviviale que `du`, `ncdu` permet de naviguer dans l'arborescence avec le clavier (flèches et retour chariot)



```
Ivan@fixe: ~/.../Bytes
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
ncdu 1.12 ~ Use the arrow keys to navigate, press ? for help
--- /home/ivan/CPGE_n/CPGE_n/CPGE_2017/Info/CoursInfo/MPII/LangageC/Bytes -----
156,0 KiB [#####] bytes.pdf
64,0 KiB [#### ] bytes.log
12,0 KiB [    ] a.out
12,0 KiB [    ] /Asup
8,0 KiB [    ] #bytes.tex#
8,0 KiB [    ] bytes.tex
4,0 KiB [    ] bytes.tex-
4,0 KiB [    ] bytes.aux
4,0 KiB [    ] bytes.nav
4,0 KiB [    ] taille.c
4,0 KiB [    ] bytes.vrb
4,0 KiB [    ] taille.c~
4,0 KiB [    ] bytes.toc
@ 0,0 B [    ] .#bytes.tex
Total disk usage: 292,0 KiB  Apparent size: 258,7 KiB  Items: 18
```

FIGURE – La commande `ncdu`

Entrer `q` pour quitter.