

TD7 : Stockage et systèmes de fichiers

1 Chemins relatifs, chemins absolus

Exercice 1 La commande `ls DIRNAME` affiche le contenu du dossier `DIRNAME`. Avec l'option `-R`, elle parcourt également tous les sous-dossiers, récursivement, et affiche leur contenu, comme illustré dans l'exemple ci-dessous.

```
% ls -R .
.:
bidule blo machin trac

./bidule:
bla bli truc

./bidule/bli:
truc

./bidule/truc:
chose

./machin:
blu chouette

./machin/blu:
truc

./machin/blu/truc:

./machin/chouette:
bla truc

./machin/chouette/truc:

./trac:
bla bli truc

./trac/bli:
truc

./trac/truc:
chose
```

- Dans notre exemple, combien y a-t-il de répertoires ? Combien de fichiers ?
- Dessinez un schéma représentant cette hiérarchie, en distinguant les fichiers et les répertoires.
- Proposez une séquence de commandes shell recréant une arborescence identique. Utilisez `mkdir PATH` pour créer les répertoires, et `touch PATH` pour les fichiers.
- Même question, mais en interdisant les arguments contenant un slash, par exemple `path/to/some/file`. Utilisez plutôt la commande `cd` pour vous déplacer.

2 Gestion de l'espace

Pour les calculs en binaire, aidez vous du tableau page suivante.

Exercice 2 On s'intéresse à un disque dur d'une capacité de 4 Tio. Combien faut-il de bits pour encoder un numéro de secteur ? On suppose que la taille d'un secteur est 512 octets.

Exercice 3 On cherche à formater en FAT une clé USB de 4Gio.

- Si on choisit une taille de bloc de 4kio (8 secteurs), combien de bits sont nécessaires pour encoder un numéro de bloc ?
- On suppose que la FAT est implémentée comme un tableau (contigu) de numéros de blocs, où chaque numéro est stocké sur un nombre entier d'octets. Quel taille occupera la FAT elle-même ?

Exercice 4 On s'intéresse à un système de fichiers unix (indexé multi-niveau) avec une taille de bloc de 2 kio et des *inodes* aux caractéristiques suivantes :

- les 12 premiers champs (numérotés 0 à 11) pointent sur des blocs de données.
- le champ n° 12 pointe vers un *bloc indirect* contenant 256 pointeurs vers des blocs de données.
- le champ n° 13 pointe vers un *bloc double-indirect*, avec 256 pointeurs vers des blocs indirects.
- le champ n° 14 pointe vers un *bloc triple-indirect*, avec 256 pointeurs vers des double-indirects.

Questions :

- Soit un fichier de 4500 octets. Combien de blocs occupe-t-il ?
- Même question pour un fichier de 567000 octets : combien de blocs sont nécessaires pour représenter ce fichier, en incluant les blocs d'indirection ?
- Quelle est la taille maximum d'un fichier sur ce système ? répondez en kio/Mio/Gio/etc, la valeur numérique exacte n'est pas intéressante.

Annexe : aide pour les calculs en binaire

Les premiers nombres entiers, notés en décimal, hexadécimal, et binaire :

| Dec | Hex | Bin |
|-----|-----|-----|
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 2 | 2 | 10 |
| 3 | 3 | 11 |
| 4 | 4 | 100 |

| Dec | Hex | Bin |
|-----|-----|------|
| 5 | 5 | 101 |
| 6 | 6 | 110 |
| 7 | 7 | 111 |
| 8 | 8 | 1000 |
| 9 | 9 | 1001 |

| Dec | Hex | Bin |
|-----|-----|------|
| 10 | A | 1010 |
| 11 | B | 1011 |
| 12 | C | 1100 |
| 13 | D | 1101 |
| 14 | E | 1110 |

| Dec | Hex | Bin |
|-----|-----|-------|
| 15 | F | 1111 |
| 16 | 10 | 10000 |
| 17 | 11 | 10001 |
| 18 | 12 | 10010 |
| 19 | 13 | 10011 |

Les premières puissances de 2, notées en décimal :

| | | | |
|--------------------|-----------------------------|------------------------------------|---|
| $2^0 = 1$ | $2^{16} = 65\ 536$ | $2^{32} = 4\ 294\ 967\ 296$ | $2^{48} = 281\ 474\ 976\ 710\ 656$ |
| $2^1 = 2$ | $2^{17} = 131\ 072$ | $2^{33} = 8\ 589\ 934\ 592$ | $2^{49} = 562\ 949\ 953\ 421\ 312$ |
| $2^2 = 4$ | $2^{18} = 262\ 144$ | $2^{34} = 17\ 179\ 869\ 184$ | $2^{50} = 1\ 125\ 899\ 906\ 842\ 624$ |
| $2^3 = 8$ | $2^{19} = 524\ 288$ | $2^{35} = 34\ 359\ 738\ 368$ | $2^{51} = 2\ 251\ 799\ 813\ 685\ 248$ |
| $2^4 = 16$ | $2^{20} = 1\ 048\ 576$ | $2^{36} = 68\ 719\ 476\ 736$ | $2^{52} = 4\ 503\ 599\ 627\ 370\ 496$ |
| $2^5 = 32$ | $2^{21} = 2\ 097\ 152$ | $2^{37} = 137\ 438\ 953\ 472$ | $2^{53} = 9\ 007\ 199\ 254\ 740\ 992$ |
| $2^6 = 64$ | $2^{22} = 4\ 194\ 304$ | $2^{38} = 274\ 877\ 906\ 944$ | $2^{54} = 18\ 014\ 398\ 509\ 481\ 984$ |
| $2^7 = 128$ | $2^{23} = 8\ 388\ 608$ | $2^{39} = 549\ 755\ 813\ 888$ | $2^{55} = 36\ 028\ 797\ 018\ 963\ 968$ |
| $2^8 = 256$ | $2^{24} = 16\ 777\ 216$ | $2^{40} = 1\ 099\ 511\ 627\ 776$ | $2^{56} = 72\ 057\ 594\ 037\ 927\ 936$ |
| $2^9 = 512$ | $2^{25} = 33\ 554\ 432$ | $2^{41} = 2\ 199\ 023\ 255\ 552$ | $2^{57} = 144\ 115\ 188\ 075\ 855\ 488$ |
| $2^{10} = 1024$ | $2^{26} = 67\ 108\ 864$ | $2^{42} = 4\ 398\ 046\ 511\ 104$ | $2^{58} = 288\ 230\ 376\ 151\ 711\ 744$ |
| $2^{11} = 2048$ | $2^{27} = 134\ 217\ 728$ | $2^{43} = 8\ 796\ 093\ 022\ 208$ | $2^{59} = 576\ 460\ 752\ 303\ 423\ 488$ |
| $2^{12} = 4\ 096$ | $2^{28} = 268\ 435\ 456$ | $2^{44} = 17\ 592\ 186\ 044\ 416$ | $2^{60} = 1\ 152\ 921\ 504\ 606\ 846\ 976$ |
| $2^{13} = 8\ 192$ | $2^{29} = 536\ 870\ 912$ | $2^{45} = 35\ 184\ 372\ 088\ 832$ | $2^{61} = 2\ 305\ 843\ 009\ 213\ 693\ 952$ |
| $2^{14} = 16\ 384$ | $2^{30} = 1\ 073\ 741\ 824$ | $2^{46} = 70\ 368\ 744\ 177\ 664$ | $2^{62} = 4\ 611\ 686\ 018\ 427\ 387\ 904$ |
| $2^{15} = 32\ 768$ | $2^{31} = 2\ 147\ 483\ 648$ | $2^{47} = 140\ 737\ 488\ 355\ 328$ | $2^{63} = 9\ 223\ 372\ 036\ 854\ 775\ 808$ |
| | | | $2^{64} = 18\ 446\ 744\ 073\ 709\ 551\ 616$ |

On rappelle également que :

- 1 kio = 1024 octets,
- 1 Mio = 1024 Kio,
- 1 Gio = 1024 Mio,
- 1 Tio = 1024 Gio,
- etc. (avec dans l'ordre : Pio, Eio, Zio, Yio)

En cas de doute sur ces unités, n'hésitez pas à demander des précisions.