

Fichiers

Arnaud Bannier
Nicolas Bodin
Aurélien Texier

1. La théorie

Exercice 1. Les modes

Quels sont les modes à utiliser pour ouvrir un fichier en :

- 1.1) lecture seule ;
- 1.2) ajout ;
- 1.3) lecture / écriture et écrasement du fichier s'il existe déjà ;
- 1.4) ajout / lecture en mode binaire.

Exercice 2. Les fonctions

- 2.1) Citez la fonction permettant d'ouvrir un fichier. Que renvoie-t-elle si l'ouverture échoue ?
- 2.2) Citez deux fonctions permettant de lire dans un fichier.
- 2.3) Citez deux fonctions permettant d'écrire dans un fichier.
- 2.4) Citez la fonction permettant de fermer un fichier.
- 2.5) Citez la fonction permettant de se déplacer à un endroit précis dans un fichier.
- 2.6) Cherchez la fonction permettant de savoir si la fin du fichier est atteinte.

Exercice 3. Une fonction pour un type de lecture

Quelle fonction utilisez vous pour lire ...

- 3.1) un fichier de configuration (mots clés + valeurs séparés par des retours à la ligne) ?
- 3.2) l'intégralité d'un fichier texte ?
- 3.3) une image au format JPEG ?
- 3.4) une page HTML ?

2. La pratique

Exercice 4. La taille ça compte

Écrivez une fonction simple renvoyant la taille du fichier passé en paramètre. Attention, vous ne devez pas utiliser de fonction de lecture.

Exercice 5. En mode barbare !

Comme vous le savez, l'ouverture d'un fichier en mode écriture supprime le contenu d'un fichier si ce dernier est présent lors de l'ouverture. L'objectif de cet exercice est de mettre en garde l'utilisateur qui ouvre un fichier existant en mode binaire.

- 5.1) Comment tester la présence d'un fichier lorsque l'on possède un chemin ? Quelle fonction utiliser ?
- 5.2) Utilisez votre réponse précédente afin d'afficher un message d'avertissement indiquant que le fichier que veut ouvrir l'utilisateur sera écrasé. Demandez-lui également s'il veut vraiment continuer. Dans le cas d'un abandon, quittez le programme.

Exercice 6. Fichiers binaires.

L'objectif de cet exercice est de vous sensibiliser à la lecture de fichiers dits binaires. À la différence d'un fichier texte (avec une extension `.txt`, `.html`, `.c` ...), la valeur des octets lus dans les fichiers binaires n'est pas interprétée en tant que caractère, uniquement en tant que valeur. Cela signifie que le mode d'ouverture est extrêmement important pour ce type de fichiers, ce que nous allons voir immédiatement.

- 6.1) Ouvrez une image sans utiliser le mode binaire.
- 6.2) Utilisez la fonction de l'exercice 4 pour allouer un tableau de caractères non signés de la même taille que celle de votre fichier.
- 6.3) À l'aide de la fonction `fread()`, stockez votre fichier dans votre tableau en le lisant octet par octet en utilisant (entre autres) une boucle `while` et la fonction `feof()`. Combien d'octets avez-vous réussi à lire ?
- 6.4) Reprenez votre code en ouvrant l'image en mode binaire et lisez à nouveau le fichier octet par octet. Un commentaire ?
- 6.5) Lisez cette fois-ci le fichier d'un seul bloc en utilisant `fread()`.
- 6.6) Regardez le fonctionnement de la fonction `clock()` associée au type `clock_t` et à la constante `CLOCK_PER_SEC` pour mesurer le temps d'exécution des lectures octet par octet et en un bloc. Comment expliquez-vous cet écart ?

3. Ça sent le sapin ...

La magie de Noël nous envahit petit à petit, le temps devient gris, les températures baissent, les cheminées fument ... À moins d'un mois de Noël, il est grand temps de créer votre sapin ! Celui-ci sera un peu particulier car il sera réalisé en ASCII-art et paramétré par un fichier de configuration passé en argument de la ligne de commande.

Ce fichier doit contenir 4 valeurs, séparées chacune par un espace :

- une première valeur pour la hauteur de votre sapin (sans le pied),
- une seconde pour la hauteur du pied,
- une troisième pour la largeur du pied,
- et une dernière représentant la proportion moyenne de boules dans le sapin. Ces boules sont générées aléatoirement.

```
      *
    **0
  *0***
*****0*
**0*****0
*0*****0****
  |||
  |||
```

FIGURE 1 – Mon beau sapin

Vous afficherez votre sapin dans la console et à l'aide de la redirection de la console (symbole >), vous pourrez stocker votre sapin dans un fichier pour l'imprimer par la suite ! Voici un exemple d'utilisation du programme qui utilise le fichier `Config.txt` pour créer un sapin qui sera stocké dans le fichier `MonSapin.txt`.

```
pantoufle@bodin:~$ ./generateur_de_sapin.exe Config.txt > MonSapin.txt
```