TD5: Lecture et Ecriture dans un fichier

Novembre 2015

Exercice 1. Buffer

Récupérez le programme `buffer.c`.

a. Après avoir redirigé la sortie standard du programme dans un fichier, lisez le temps que le programme met à s'exécuter.

b. Tester les différents mode du buffer en modifiant les commentaires. Dans quel mode semble être paramétré le flux `stdout` par défaut ?

Exercice 2. Commande wc

Programmer la commande du terminal `wc -c` qui donne le nombre de caractères d'un fichier texte passé en argument. Le programme sera appelé de la manière suivante :

```
./a.out fichier
```

Exercice 3. Ça se dégrade

Dans cet exercice, vous devrez créer une image au format ppm ASCII. Un tel fichier contient 3 lignes d'en-tête :

— P3 sur la première ligne ce qui correspond au type de fichier (ppm en mde texte),
— la largeur de l'image et la hauteur de l'image séparées par un espace,
— le nombre de couleurs.

Ensuite les pixels de l'image sont donnés ligne par ligne de gauche à droite. Un pixel est représenté par trois nombres entre 0 et 255 qui représentent son intensité en couleur pour les trois couleurs de base, rouge, vert et bleu. Toutes ces valeurs sont séparées par un retour à ligne dans un fichier ppm. Voir `http://fr.wikipedia.org/wiki/Portable_pixmap` pour la documentation sur ce formats.

a. Créer une image de 256×256 où le point en haut à gauche est noir, celui en haut à droite est bleu, celui en bas à gauche est rouge et celui en bas à droite est magenta. Tous les autres points seront des dégradés de rouge et de bleu proportionnels à leurs coordonnées.

b. Quelle taille fait le fichier ?

c. Visualiser les images.

Exercice 4. Cryptographie

Le but de cet exercice est de cacher du texte dans une image.

Pour cacher une chaîne de n caractères dans une image, on va modifier les n premiers pixels (r, v, b) de l'image de la manière suivante : on coupe la valeur du caractère lu en trois ; ainsi, le caractère a représenté par 11000001 se coupe en $11 = 3$, $000 = 0$ et $001 = 1$. Ensuite on remplace les trois bits de poids faible, c'est-à-dire les trois derniers bits, de r, v et les deux derniers bits de b par ces valeurs. Par exemple si $r = 10111101$, $v = 01010110$ et $b = 00010011$, alors le pixel modifié par le caractère a vaut $r = 10111111$, $v = 01010000$ et $b = 00010001$.

a. Quelle formule permet d'obtenir le pixel modifié (r', v', b') en fonction de la valeur ASCII du caractère lu ?

b. Ouvrir l'image `chat` fournie sur ecampus avec un visionneur d'image. L'ouvrir ensuite avec un éditeur de texte comme Geany.

c. Compilez le fichier `decrypt.c`, puis exécutez le sur le fichier `chatCrypte` de la manière suivante :

```
./a.out chatCrypte
```

Que dit le chat ? Voit-on une différence entre les images `chat` et `chatCrypte` ? Comprendre le code de `decrypt.c`.

d. Écrire le programme `crypt.c` qui affiche dans le terminal le fichier avec le message crypté. Celui-ci sera appelé par la commande suivante dans le terminal :

```
./a.out chat "message a cacher"
```

Si vous voulez l'écrire dans un fichier pour pouvoir le lire, vous redirigerez la sortie standard de la manière suivante :

```
./a.out chat "message a cacher" >fichierCrypte
```

Vous vérifierez que votre programme est correct en décryptant le fichier que vous avez obtenu.

Dans les 3 exercices qui suivent, vous répondrez aux questions suivantes :

1. Quelle taille fait le fichier généré ?
2. Dans le shell lancer la commande `file` avec le nom de fichier en argument, quel est le résultat ?
3. Ouvrir le fichier avec geany, que lisez-vous ?

Exercice 5. Mon premier fichier, que du texte, ça c'est sûr.

Écrire dans un fichier en mode texte :
— Sur la première ligne les lettres de l'alphabet en minuscule sans espace ;
— Sur la deuxième ligne les lettres de l'alphabet en majuscule sans espace ;
— Sur les lignes suivantes les nombres de 0 à 999 inclus en affichant chaque nombre sur 4 caractères et en revenant à la ligne tous les 10.

Exercice 6. Mon deuxième fichier que du binaire ?

Écrire dans un fichier en mode binaire sans séparateurs :
— Les lettres de l'alphabet en minuscule ;
— Les lettres de l'alphabet en majuscule ;
— Écrire le caractère '`\n`' ;
— Écrire les nombres de 0 à 999.

Exercice 7. Est-ce du binaire ou du texte ?

Écrire dans un fichier en mode binaire sans séparateurs :
— Les lettres de l'alphabet en minuscule ;
— Les lettres de l'alphabet en majuscule ;
— Écrire le caractère '`\n`' ;

Exercice 8. J'écris donc je lis

Choisissez le mode (binaire ou texte) qui vous semble le plus adapté pour effectuer les opérations suivantes :

a. Dans un premier programme, générer 10^6 entiers aléatoires positifs et les écrire dans un fichier `a.txt`.

b. Dans un deuxième programme, lire le fichier `a.txt` et écrire les nombres dans `pair.txt` ou `impair.txt` en fonction de leur parité.

Exercice 9. Patron de boîte.

On reprend le cadre de l'exercice 3.

a. Créer un patron de cube permettant de visualiser les différentes couleurs. Les huit sommets du cube doivent être noir, rouge, vert, bleu, jaune, cyan, magenta, blanc. Chaque face devant présenter le dégradé sur deux des trois composantes RGB.

b. Écrire ce patron dans un fichier au format ppm binaire.