

Programmation Système

TP 1 : Rappels

Juliusz Chroboczek
Khouloud Zine Elabidine

17 et 24 septembre 2012

L'objectif de ce TP est d'écrire une commande `cat2` qui concatène les fichiers qui lui sont donnés en argument dans un fichier destination (comprendre « cat to »).

Question 1. Écrivez une commande `cat2` telle que `cat2 file` crée le fichier `file` de taille nulle. S'il existe déjà elle effacera son contenu. On prendra soin de gérer les éventuelles erreurs. (*open, close, perror, exit*)

Question 2. Modifiez `cat2` pour qu'il concatène dans ce fichier les divers fichiers qui lui sont donnés à la suite sur la ligne de commande. Par exemple `cat2 file a b c` doit concaténer les fichiers `a`, `b` et `c` dans le fichier `file`. (*read, write*)

Question 3. Faites en sorte que votre commande donne des droits d'accès au fichier `file` qui soient aussi restrictifs que ceux de tous les autres fichiers. Par exemple, si les droits d'accès au fichier `a` sont `rw---` et les droits du fichier `b` sont `r-r-r-`, le fichier créé par concaténation ne devra autoriser que la lecture pour le propriétaire : `r----`. (*stat*)

Question 4. Modifiez `cat2` pour que `cat2 fic dir` concatène tous les fichiers contenus dans `dir` lorsque celui-ci est un répertoire. (*opendir, closedir, readdir, optionnellement openat*)

Question 5. Faites en sorte que la taille du tampon utilisée par votre commande soit paramétrable : une option permettra à l'utilisateur de la choisir. Par exemple, la commande `cat2 -b 10 fic a b c` utilisera un tampon de 10 octets. (*malloc, getopt, strtol*)

Question 6. Ajoutez une option `-n` pour numéroter les lignes de chaque fichier lu. (*strchr*)

Question 7 (S'il vous reste du temps). Implémentez la commande `tac(1)`, qui concatène les fichiers qui lui sont passés en paramètre et les affiche en inversant l'ordre des lignes. Votre programme s'exécute-t-il en espace constant ?