

Exercice 1

Difficulté : 34 points-virgules

On souhaite rédiger un programme `rotation` pour copier un ou plusieurs fichiers indiqués par leur chemin en modifiant leur contenu de façon que les n premiers octets de chaque fichier original soient placés à la fin du fichier copié. Le fichier copié doit porter le même nom complété par le suffixe « `.rot` » et être placé dans le même répertoire que le fichier original.

Le programme doit admettre la syntaxe suivante :

```
rotation n f1 ... fn
```

Par exemple :

```
> pwd
/home/truc
> cat /tmp/toto
hello
world !
il fait beau
> ./rotation 6 /tmp/toto /tmp/titi
> cat /tmp/toto.rot
world !
il fait beau
hello
```

les 6 premiers caractères du fichier original sont à la fin

Pour rédiger votre programme, il est impératif de respecter les contraintes suivantes :

- vous ne devez utiliser que les primitives système (ou assimilées comme telles); vous pouvez toutefois utiliser les fonctions de bibliothèque pour les affichages ou les manipulations de chaînes de caractères;
- pour des raisons d'efficacité, vous ne ferez pas d'appels redondants à des fonctions lentes (primitives système ou autres);
- toujours pour des raisons d'efficacité, vous effectuerez tant que possible des lectures ou d'écriture par quantité de `MAXBUF` octets que vous définirez à 4096 octets (si vous voulez savoir pourquoi, essayez de définir cette valeur à 1 octet);
- pour des raisons de simplicité, vous limiterez la taille du chemin des fichiers créés à la constante `CHEMIN_MAX` que vous définirez à 128 octets : un chemin plus long doit être considéré comme une erreur;
- vous vérifierez soigneusement les débordements de tableau (vous pouvez notamment utiliser la fonction de bibliothèque `snprintf` pour contrôler la taille de chaînes complexes);
- votre programme doit retourner un code de retour nul (`exit(0)`) si tout s'est déroulé sans erreur ou un code de retour non nul (`exit(1)`) si une erreur a été rencontrée;
- si votre programme est appliqué avec un nombre d'arguments incorrect, il doit afficher un message de la forme : "usage: rotation n f₁ ... f_n".
- vous apporterez un soin particulier à la mise en forme de façon à rendre un code lisible et commenté à bon escient. Référez-vous au document « Conseils pour réussir vos TP et projets » mis à votre disposition sur Moodle et, si besoin, utilisez l'utilitaire `clang-format` avec la configuration donnée dans ce document;
- votre programme doit compiler avec les options `-Wall -Wextra -Werror -pedantic` sur `gcc` version 9.4 minimum (la version disponible sur la machine `turing.u-strasbg.fr`). Alternativement, vous pouvez utiliser l'image Docker `pdagog/refc` (version de `gcc` 13.2) Les programmes qui ne compilent pas au moins sur `turing` avec ces spécifications **ne seront pas examinés**.

Un script de test est mis à votre disposition sur Moodle. Celui-ci exécute votre programme sur des jeux de tests qui serviront de base à l'évaluation de votre rendu. La commande suivante permet de lancer les tests :

```
sh test1.sh.
```

Vous devrez rendre sur Moodle un *unique* fichier nommé `rotation.c`.

Cet exercice est **individuel**. On rappelle que la copie ou le plagiat sont sévèrement sanctionnés.