

## INF203 - Exercices semaine 7

### Arguments de la ligne de commande, fichiers

#### Exercice 1 :

1. Écrire un programme qui affiche la somme des valeurs entières qui lui sont données en arguments.
2. Écrire un programme qui prend trois arguments et, si la valeur du second argument est :
  - `+` : affiche la somme des valeurs représentées par les arguments 1 et 3.
  - `-` : affiche la différence entre les valeurs représentées par les arguments 1 et 3.
  - `*` : affiche le produit des valeurs représentées par les arguments 1 et 3.
  - `/` : affiche le quotient de la division de la valeur représentée par l'argument 1 par celle représentée par l'argument 3.

#### Exercice 2 :

Le but de cet exercice est d'écrire une commande similaire à `wc` :

1. Écrire une fonction qui compte le nombre de caractères lus depuis un flux passé en argument et renvoie ce nombre.
2. Écrire un programme qui ouvre le fichier dont le nom lui est donné en argument et affiche le nombre de caractères qu'il contient. En cas d'erreur d'ouverture, votre programme devra afficher un message approprié.
3. Modifiez votre programme pour qu'il accepte aucun ou plusieurs arguments. Si aucun argument n'est donné, il devra lire depuis l'entrée standard. Sinon, pour chacun des arguments qui lui sont donnés, il devra afficher l'argument, ouvrir le fichier de nom correspondant et afficher le nombre de caractères qu'il contient. En cas d'erreur d'ouverture, votre programme devra afficher un message approprié.
4. Écrire une fonction qui compte le nombre de lignes lues depuis un flux passé en argument et renvoie ce nombre.
5. Même question pour le nombre de mots (*indication* : la directive `%s` de `scanf` correspond à la lecture d'un mot).
6. Modifier le programme de la question 3 pour tenir compte des options : si le premier argument est `-c`, `-l` ou `-w`, le programme devra respectivement afficher le nombre de caractères, lignes ou mots depuis sa ou ses entrées.

#### Exercice 3 :

Le but de cet exercice est d'écrire une commande similaire à `tail` :

1. Écrire un programme qui affiche les 10 dernières lignes que contient un fichier dont le nom est donné en argument.
2. Ajouter une option `-lines` prenant un argument entier indiquant le nombre de lignes à afficher.
3. Modifier votre programme pour accepter un nombre quelconque d'arguments et lire depuis l'entrée standard si aucun argument n'est donné ou depuis chacun des fichiers de nom donné en argument sinon.