Master IC - Pierre Pompidor
Exemple de script Système en Perl

Le but de ce TP est de vous initier au langage Perl qui est un langage assez abscons mais délicieux, en vous donnant un exemple de **script système** tout en vous faisant manipuler **emacs**.
Vous allez donc essayer d'écrire dans un premier temps un script qui dit "bonjour" notamment pour ceux qui n'auraient jamais eu une poupée qui faisait cela (je l'ai toujours pour ma part regretté), et dans un second temps un script qui suivant l'option passée (a ou p) :

- soit liste les `alias` mémorisés dans le fichier de configuration (.bashrc) ;
- soit indique les répertoires ajoutés à la variable d'environnement `PATH`.

## Ecrivez un script perl qui affiche "Bonjour !"

La première étape est d'écrire un "Hello world", donc saisissez le code suivant sous emacs (fichier `bonjour.pl`) :

```
#!/usr/bin/perl
print "Bonjour !\n";
```

Pour l'exécuter n'oubliez pas de lui donner le droit d'exécution, puis :
- si vous avez rajouté le répertoire courant (`.`) dans votre `PATH` : `bonjour.pl`
- sinon (mais c'est déshonnorant) : `./bonjour.pl`

## Modifiez ce script pour qu'il affiche "Bonjour Machin !"

Maintenant, passons un paramètre au script ; appel :
`bonjour.pl VotrePrenom`
Ce paramètre sera stocké dans un tableau de paramètre appelé `@ARGV` et dont le premier élément (c'est à dire le premier paramètre passé au script) est appelé `$ARGV[0]` → limpide !

```
#!/usr/bin/perl
print "Bonjour ", $ARGV[0], "!\n";
```

**Et puis passons aux choses sérieuses, écrivez le script waouhh.pl qui liste vos alias ou indique les répertoires ajoutés au PATH**

L'algorithme :

```
Si le paramètre est 'a' ou 'p' :
   Ouvrir le fichier .bashrc
   Si le fichier a été ouvert avec succès
       recopier le contenu du fichier dans un tableau en mémoire centrale
       Si le paramètre est 'a'
          Pour toutes les lignes du fichier
               filtrer toutes les lignes contenant alias
               afficher la ligne de commande ... est devenue ...
          FinPour
       Sinon
          Pour toutes les lignes du fichier
               filtrer toutes les lignes contenant PATH=
               afficher les répertoires supplémentaires
          FinPour
       FinSi
       fermer le fichier
   FinSi
Finsi
Sinon Message d'alerte
```

Puis pensez à une gestion plus souple des options (ex. : `waouhh.pl -ap`)
Voici quelques informations :

- pour faire un test :
  `if (expression conditionnelle) { instructions }`
- pour appliquer une expression régulière :
  `if ($variable =~ /expression/) { instructions }`
- pour ouvrir un fichier : `open descripteur, "fichier";`
- pour recopier le contenu d'un fichier en mémoire centrale :
  `@tableau = <descripteur>`
- pour parcourir un tableau :
  `foreach $variable (@tableau) { instructions }`