AUBARET Auriane TD10

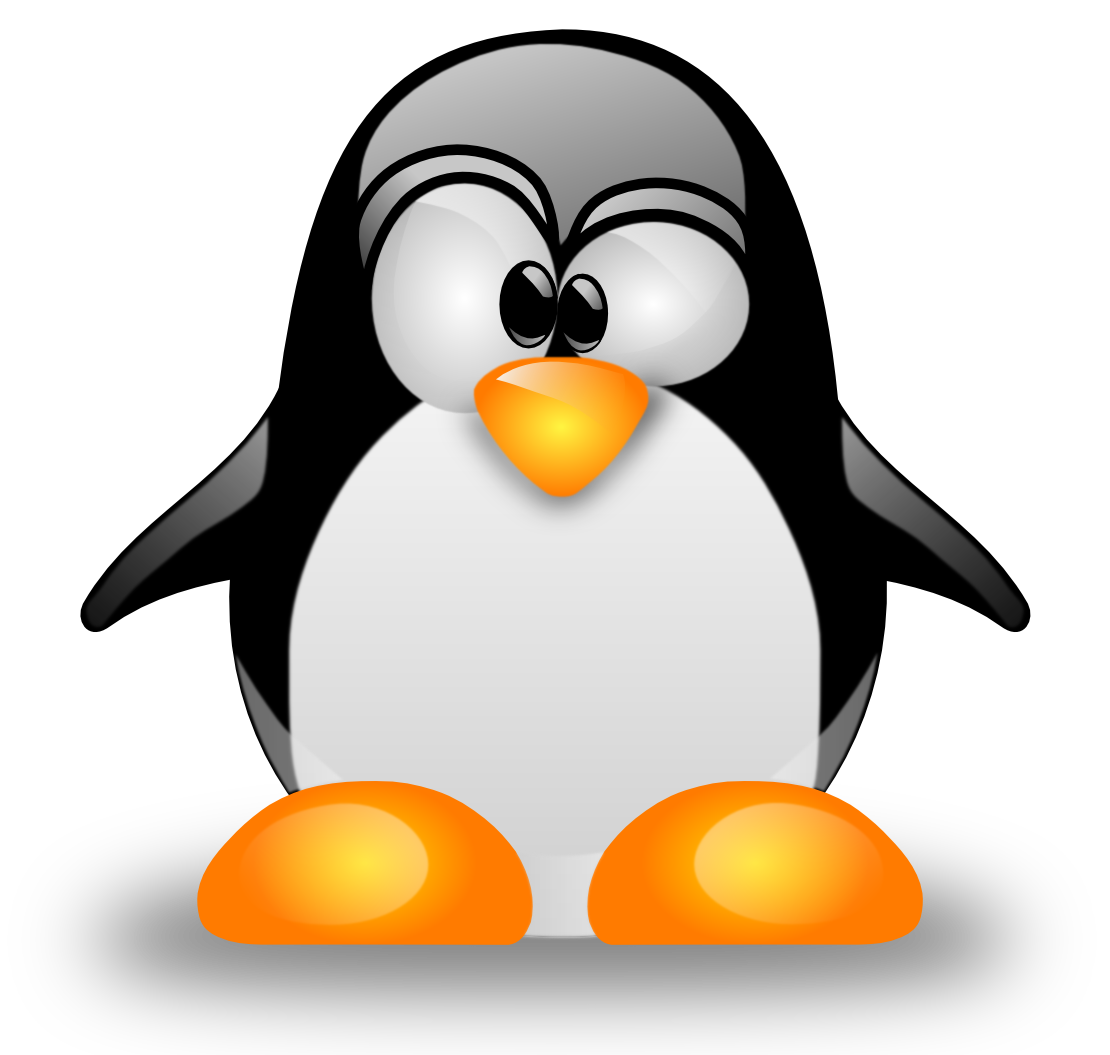
LOBA Alexandre TD05

ALBY Antoine TD07

ING2 – TEAM A

18/05/2017

Mini projet Linux



Sommaire

Présentation du sujet…………………………………………………………2

1. Organisation de l’équipe………………………………............3
2. Présentation des fonctions
3. Bibliothèques inclues……………………………………………………4
4. Fonctions prédéfinies utilisées………………………………………4
5. Fonctions implémentées………………………………………………5
6. Conclusion………………………………………………………………6
7. Code source……………………………………………………………7

Présentation du sujet

Le travail qui nous a été demandé était de réaliser un mini-Shell personnalisé comprenant quelques fonctionnalités de basique du Shell. Nous avons travaillé en particulier sur des fonctionnalités de gestion de fichiers.

C’est un projet entièrement réalisé sous Linux mais avec un code en langage C.

Nous avons réalisé les 7 fonctionnalités suivantes :

* Terminaison
* Changement de répertoire
* Affichage du répertoire et listage du contenu
* Affichage d’un prompt personnalisé
* Listage du contenu d’un répertoire
* Suppression d’un fichier
* Renommer un fichier

Pour toutes les fonctions, un message d’erreur doit apparaitre pour toute saisie de commande incorrecte ou de fichiers/ répertoires inexistants.

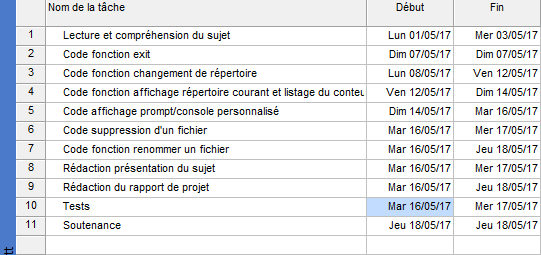
Il était obligatoire d’utiliser l’outil de Versionning (Git + Bitbucket) au cours de son projet.

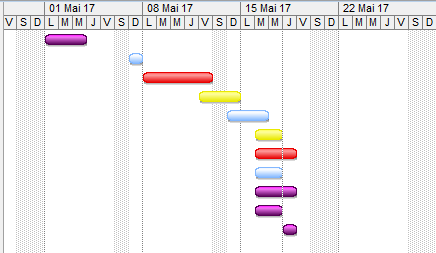
1. Organisation de l’équipe

Nous nous sommes réparti les tâches de la façon suivante, que vous pouvez visualisez avec les Diagrammes de Gant ci-dessous.

Les couleurs sont les suivantes :

* Antoine : Rouge
* Alexandre : Bleu
* Auriane : Jaune





1. Présentation des fonctions
2. Bibliothèques inclues

Nous avons décidé d’inclure les bibliothèques suivantes dans notre projet :

* stdio.h : bibliothèque standard pour la définition des fonctions en C.
* stdlib.h : bibliothèque pour l’allocation mémoire, conversion chaînes de caractères, etc.
* signal.h : contrôle des signaux en C.
* unistd.h : bibliothèque standard
* string.h : pour les chaînes de caractères.
* erno.h : bibliothèque contenant un ensemble de code d’erreurs.
* dirent.h : pour l’utilisation des dossiers.

1. Fonctions prédéfinies utilisées

La fonction exit

C’est une fonction qui permet de sortir du Shell. Elle est relativement simple, c’est pour cela que nous avons décidé de l’utiliser prédéfinie.

La fonction d’affichage d’un prompt personnalisé

Cette fonction permet de changer de prompt, c’est-à-dire de console. Elle prend en paramètre un double pointeur sur la console.

On déclare une chaîne de caractère correspondant au nom du répertoire courant. S’il est effectivement différent du prompt, on change de console, dans le cas contraire contraire un message d’erreur s’affiche à l’écran.

La fonction pour supprimer un fichier

Cette fonction permet d’effectuer la suppression d’un fichier.

Elle prend en paramètre un pointeur sur le nom du fichier que l’utilisateur souhaite supprimer. On effectue l’action et si celle-ci ne peut pas se faire correctement un message d’erreur est affiché à l’écran.

1. Fonctions implémentées

La fonction pour changer de répertoire

Cette fonction permet à l’utilisateur de switcher entre deux répertoires.

Elle prend en paramètre un pointeur sur une chaîne de caractère. Celle-ci correspond au nom du répertoire souhaité par l’utilisateur.

Si jamais l’utilisateur veut accéder à un répertoire qui n’existe pas, un message d’erreur s’affiche à l’écran.

La fonction pour renommer un fichier

Cette fonction permet de changer le nom d’un fichier, c’est-à-dire de le renommer.

Elle prend en paramètre un pointeur sur une chaîne de caractère correspondant au nom du fichier que l’utilisateur souhaite renommer.

Par la suite, on déclare et initialise deux pointeurs sur deux chaînes de caractères ainsi qu’un entier. On enregistre la commande de l’utilisateur.

Ensuite, si le nom des deux fichiers sont bien différents, on change le nom du fichier en question. Si ce n’est pas le cas, on indique à l’utilisateur qu’il est impossible d’effectuer cette action puisque le nouveau nom est déjà celui du fichier.

La fonction pour afficher le répertoire courant et lister son contenu

Cette fonction permet d’afficher le contenu d’un répertoire ainsi que lister son contenu.

Elle ne prend aucune variable en paramètre. On déclare différentes variables : une chaine de caractère correspondant au nom du répertoire en question, un pointeur sur une chaine de caractère, en entier initialisé à 0 dans le cas d’un répertoire vide (puisque l’on n’a pas de booléen en C). Si le répertoire existe, on affiche son contenu, dans le cas contraire un message d’erreur apparait à l’écran.

1. Conclusion

Ce projet a été très enrichissant. Nous avons pu découvrir de façon plus approfondi les différentes fonctionnalités de Linux. Nous avons également pu mettre en pratique les cours dispensés ce semestre, en particulier ceux sur la gestion des processus, la gestion des signaux et le Shell.

Nous avons réutilisé des fonctionnalités du langage C, apprise l’année précédente et parfois oublié. Le langage C étant un langage de base, il est très intéressant pour nous de bien le maitriser.

Il a aussi été intéressant de travailler sur un OS de bas niveau, contrairement à ceux que nous avons l’habitude d’utiliser.

1. Code source







