

## Travaux Pratiques d'informatique

# Algorithmique et langage C

## 1 Introduction à Linux et à la programmation

### 1.1 Connection, changement du mot de passe et activation de votre mail

Lors de la première séance de TP, vous allez découvrir votre environnement de travail.

Vous devez tout d'abord vous connecter en utilisant le login et le mot de passe par défaut qui vous a été fourni par l'enseignant. Ensuite, vous devrez modifier votre mot de passe. N'oubliez pas que :

1. Votre mot de passe est strictement personnel.
2. Vous ne devez jamais le communiquer (ni à l'équipe enseignante, ni à l'administration, ni à vos proches).
3. Si vous perdez votre mot de passe, prenez contact avec l'équipe système qui vous redonnera un nouveau mot de passe.
4. Enfin, les environnements Linux et Windows sont déconnectés... vous devez donc changer vos mots de passe sous les deux environnements pour éviter que votre mot de passe par défaut reste actif sur l'un des deux environnements.

La procédure de changement de mot de passe (sous Linux et Windows) est automatique lors de la première connection.

Faites bien attention aux touches que vous utilisez, vérifiez que le clavier n'est pas verrouillé (touche majuscules/minuscules).

Pour activer votre compte mail, connecter vous sur la page web <http://www.annuaire.u-psud.fr/> puis choisissez l'onglet *Activation étudiant* dans la rubrique *Services Liés*

Lorsque ces deux opérations sont effectuées, vous pourrez débiter votre TP.

### 1.2 Commandes élémentaires et structuration de votre espace de travail

Voici une liste de commandes qui vous permettront d'exploiter au mieux votre espace de travail via la console.

Commande	Action
<b>mkdir</b> DIRECTORY	Création d'un nouveau répertoire nommé DIRECTORY
<b>cd</b> DIRECTORY	Entrer dans le répertoire DIRECTORY
<b>cd</b> ~	Retourner dans votre répertoire principal (à la <b>racine</b> de votre compte)
<b>pwd</b>	Afficher à quel niveau de l'arborescence vous vous situez
<b>ls</b>	Lister le contenu du répertoire courant
<b>cp</b> FICHIER_SOURCE DIRECTORY	Copier FICHIER_SOURCE dans DIRECTORY
<b>cp</b> FICHIER_SOURCE DIRECTORY/FICHIER_DESTINATION	Copier FICHIER_SOURCE dans DIRECTORY/FICHIER_DESTINATION
<b>mv</b> SOURCE DESTINATION	Déplacer/renommer un fichier SOURCE dans une DESTINATION
<b>rm</b> FICHIER	Supprimer le FICHIER

Nous utiliserons systématiquement la console pour compiler (transformer le texte du programme en code exécutable) et exécuter le programme obtenu.

La première opération à réaliser est de structurer votre espace de travail pour pouvoir y organiser vos données. Vous allez pour cela créer un **répertoire** nommé *tps\_c* en utilisant la commande suivante : `mkdir tps_c` Puis changer de répertoire pour aller dans le répertoire *tps\_c* : `cd tps_c`, et enfin y créer un nouveau répertoire *tp1*.

Vous venez de structurer votre espace de travail de manière cohérente. Cela vous permettra de retrouver facilement vos TP par la suite.

Dans la mesure du possible, essayez systématiquement de structurer vos données de la manière la plus cohérente possible.

### 1.3 Votre premier programme

Afin de pouvoir écrire et tester votre premier programme, vous devrez commencer par choisir un éditeur de textes pour saisir le texte de votre programme.

Vous avez à votre disposition de nombreux éditeurs, du plus “rudimentaire” **vim**, jusqu’aux plus évolués **Anjuta**, **KDevelop** ou **Eclipse** qui sont des IDE <sup>1</sup>, en passant par des éditeurs de texte plus “classiques” tels que **Kate**, **Kedit**, **gedit**, ...

La plupart de ces éditeurs proposent la coloration syntaxique, ainsi que l’indentation automatique du code. L’important est de choisir un éditeur qui vous conviennent et que vous arrivez bien à utiliser (raccourcis claviers, fonctionnalités diverses, ...).

**On s’interdira dans ce module l’utilisation des IDE**, afin d’apprendre à compiler correctement. Vous utiliserez l’éditeur de texte de votre choix.

Après avoir choisi un éditeur, vous pourrez écrire votre premier programme *helloWorld.c*.

Ouvrez un nouveau fichier, saisissez le code “Hello World” vu en cours, puis sauvegardez le fichier dans le répertoire `~/tps_c/tp1/helloWorld.c`.

Ensuite, dans une console, placez vous dans le répertoire `~/tps_c/tp1` et exécutez la commande permettant de compiler le programme pour obtenir une version exécutable :

```
gcc -o helloWorld helloWorld.c
```

Si la compilation s’est bien déroulée, vous avez obtenu un programme exécutable qui se trouve dans votre répertoire courant. Vous pouvez simplement l’exécuter en utilisant la commande : `./helloWorld`

Vous pouvez ajouter deux options de compilations qui vous permettront de détecter des erreurs dans le texte de vos programmes :

- **-Wall** : pour activer tous les **warnings**. Dans la mesure du possible, il faut supprimer corriger le texte du programme afin de ne plus avoir de warnings. En cas de doute, demandez l’aide de votre enseignant.
- **-ansi** : pour activer la vérification de la norme ANSI<sup>2</sup>. Ceci vous permettra de garantir que le texte de votre programme respecte bien la norme ANSI et vous garantira ainsi la portabilité de votre programme sur d’autres environnements respectant également cette norme (Windows, Mac, Unix, ...).

Vous pouvez recompiler votre programme avec ces options de la manière suivante :

```
gcc -o helloWorld helloWorld.c -Wall -ansi
```

Vous disposez maintenant des bases suffisantes pour pouvoir écrire des programmes en C, les compiler et les exécuter. Afin de bien maîtriser tous les concepts et commandes vus dans ce TP, il faut pratiquer !!!

## 2 Codage des exercices vu en tds

Vous programmerez les exercices qui ont été vu et résolus en travaux dirigés.

1. IDE : **I**ntegrated **D**evelopment **E**nvironment (Atelier de développement intégré)

2. American National Standards Institute