

# NSY103 - Linux : principes et programmation

🔴 Valide le 14-04-2020

## Présentation

### Prérequis

Avoir des bases sur le fonctionnement des systèmes d'exploitation (cette UE intervient dans des diplômes et certifications de niveau supérieur à Bac + 2).

Savoir lire un programme écrit en langage C

### Objectifs pédagogiques

Développement d'applications multiprocessus entrainées et réparties sur un système LINUX.

### Compétences

Savoir développer une application multi processus utilisant des outils de communication et de synchronisation en C sous Linux/Unix.

## Programme

### Contenu

#### Introduction générale

- Rappels sur le fonctionnement des systèmes informatiques.
- Systèmes mono et multi processeur, centralisés et répartis
- Installation d'un système Linux et d'un environnement de développement C (directement ou dans une machine virtuelle)
- Fonctionnement général du système Linux

#### Gestion de processus

- Processus sous linux
- Réalisation d'un appel système
- Threads
- Gestion de l'espace d'adressage
- Ordonnancement sous Linux mono et multiprocesseur
- Langage de commandes associé

#### Communication centralisée et Synchronisation de processus

- tubes nommés et anonymes, files de messages
- segments de mémoire partagée
- signaux
- Section critique, sémaphores, problèmes classiques.
- exemple d'interblocage (par exemple avec les tubes nommés)

#### Systèmes de fichiers

- Interfaces des systèmes de fichiers et implémentation
- programmation
- langage de commande.
- fichiers spéciaux
- gestion des droits

#### Scripts

- les notions de bases (variables, boucles, conditionnelles)
- variables d'environnement



### Code : NSY103

6 crédits

#### Responsabilité nationale :

EPN05 - Informatique / Joëlle DELACROIX-GOUIN

#### Contact national :

EPN05 - Informatique

2 rue Conté

33.1.13

75003 Paris

01 40 27 27 02

Emmanuelle BIAR

[emmanuelle.biar@lecnam.net](mailto:emmanuelle.biar@lecnam.net)

## Systèmes distribués

- Structure des réseaux et structure des systèmes répartis.
- Programmation socket
- Introduction au RPC

**Des travaux pratiques permettent d'introduire les notions liées au langage C et de mettre en oeuvre les primitives du système liées au processus et à la communication entre processus**

**Les différents items sont abordés coté utilisateur (programmation) et coté système (implémentation). Par exemple, on peut détailler le fonctionnement interne de quelques appels système et la présentation des structures de données et algorithmes du noyau.**

## Modalités de validation

- Examen final
- Projet(s)

## Description des modalités de validation

L'évaluation de première et deuxième session est axée autour :

1/ de trois devoirs maison dont le dernier est un projet de mise en oeuvre des outils de communication donné à réaliser aux auditeurs. Ce projet conduit à la spécification et programmation d'une application multiprocessus communicant via les outils étudiés (tubes, MSQ, sockets, etc...). On donnera tout d'abord une spécification centralisée puis on demandera à l'étendre en réparti

**Ces trois devoirs notés chacun sur 5 points comptent pour un quart de la note finale de première et deuxième session.**

2/ d'un examen écrit de deux heures comptant pour  $\frac{3}{4}$  de la note finale.

3/ De tps qui peuvent être notés et apporter des points bonus.

## Bibliographie

Titre	Auteur(s)
Linux : programmation système et réseau, Dunod. Quatrième édition	Joëlle Delacroix