

## TP2 Système de fichiers et processus

nicolas.ollinger@univ-orleans.fr

20 octobre 2017

*Contrôle continu : Sont à rendre selon les modalités détaillées sur celene, une archive contenant l'utilitaire **runas** de l'exercice 6 accompagné d'un **Makefile**.*

### 1 Système de fichiers

**Exercice 1.** Écrire un programme **ls** qui se comporte comme l'utilitaire POSIX **ls -li**. On portera un soin particulier aux détails : chemins contenus dans les liens symboliques, type de fichier (champ **st\_mode** de **struct stat**), etc.

**Exercice 2.** Écrire un programme **df** qui affiche des informations sur le système de fichier associé au fichier passé en paramètre : nombre de blocs, blocs utilisés, blocs disponibles, capacité, etc.

**Exercice 3.** Écrire un programme **stat** qui affiche l'ensemble des méta-données associées au fichier passé en paramètres.

### 2 Processus

**Exercice 4.** Écrire la fonction de bibliothèque **int system(const char \*command)**; qui exécute la ligne de commande passée en paramètre et renvoie la valeur de retour de cette exécution.

**Exercice 5.** Écrire un programme **time** qui chronomètre le temps d'exécution du programme passé en paramètre ainsi que son temps d'activité utilisateur et noyau.

**Exercice 6.** Écrire un programme **runas name program arg1 arg2 ... argN** qui exécute le programme **program arg1 arg2 ... argN** avec **name** comme valeur de **argv[0]** au lieu de **program**.

### 3 Fonctions diverses

**Exercice 7.** Complétez votre bibliothèque d'utilitaires POSIX en écrivant des programmes : **echo**, **pwd**, **date**, **true**, **false**, **sleep**, **touch**, **find**, etc.