

# Ing 1 - Système d'exploitation

## 1 Préambule

Vous devrez répondre aux questions de cours sur papier. Les réponses que vous apporterez pour les scripts shell se fera elle de manière électronique. Vous placerez ces fichiers dans votre répertoire de rendu : `renduse`.

## 2 Réflexions sur le cours - 8 points

Pour chaque réponse, il vous est demandé d'argumenter. Une réponse non argumentée ne pourra donner lieu à une quelconque gratification !

- 1 pt – Qu'est ce qu'un thread ?
- 1 pt – Libérer la mémoire d'un processus quand ce dernier passe dans l'état zombie a-t-il un sens ?
- 2 pt – Un attribut essentiel des fichiers est la longueur. Pourquoi le système d'exploitation doit-il y faire attention ?
- 2 pt – Sous Windows, il est facile de savoir quand archiver un fichier, puisque chaque fichier possède un bit d'archivage. Sous Unix, ce bit n'est pas présent. Comment les programmes de sauvegarde d'Unix ont-ils la connaissances des fichier à sauvegarder ?
- 2 pt – Sous Unix, les *i-nodes* sont stockés au début du disque. Une autre solution serait de stocker ces *i-node* au début du premier bloc de chaque fichier. Quel serait les avantages et inconvénients de cette méthode ?

## 3 Script shell - 12 points

```
-r : récursif  
-h : aide  
--help : aide  
-f : fichier
```

FIG. 1 – Fichier d'options

1. Soit le fichier *exemple*<sup>1</sup> de la figure 1. Écrire un script qui utilise un fichier de ce type, et qui traite les arguments de la ligne de commande en fonction des indications marquées dans le fichier. Par exemple, supposons que le script s'appelle `argum`. L'exécution de la commande `argum -f` affichera "fichier"; `argum --help` affichera "aide"; `argum -t` affichera "option non traitée".
2. Écrire un script qui permet de générer une séquence de nombres entiers consécutifs. Vous ne devez pas utiliser la commande `seq`. Ce script traitera les options suivantes :

```
-d nb : nombre de départ de la séquence
-f nb : nombre de fin de la séquence
-p nb : incrément entre chaque nombre entier
-h, --help : affiche l'aide
```

Si une ou les deux options `-f`, ou `-p` ne sont pas spécifiées, alors vous considérerez qu'elles valent 1.

3. Écrire un script qui demande un nom et un prénom. Puis qui pour chaque script, supposé bien rédigé, d'un répertoire passé en paramètre, ajoute en deuxième ligne le commentaire suivant :

```
# Auteur : nom prénom
```

---

<sup>1</sup>Ce qui veut dire que le fichier de la figure 1 est un exemple. Vous devez donc prévoir plus de possibilité...