

Rédigé par : équipe pédagogique du cours de Syst. D'expl.

Ref : *ING1-SE-EXAMEN*

A l'intention de : Etudiants d'ING1

Créé le : 17/01/2012

### 1. Script shell (5 pts)

Question 1 : Écrire un script shell qui prend en paramètre un fichier formaté comme indiqué ci-dessous et affiche tous les livres d'un auteur passé en paramètre.

Question 2 : Prendre en compte l'option -h

Question 3 : Tester tous les cas possibles

#### Exemple de fichier (bibliotheque):

Leon Shklar & Rich Rosen @ Web Application Architecture

Bernard Desgraupes @ LaTeX : apprentissage, guide et référence

Vipul Kashyap & Leon Shklar @ Real World Semantic Web Applications

Ernest E. Rothman & Brian Jepson & Rich Rosen @ Mac OS X for Unix Geeks (Leopard)

Bernard Desgraupes @ Passeport Pour Unicode

Exemple d'appel (en gras l'appel, en italique la réponse du script) :

#### **./script bibliotheque Desgraupes**

*Desgraupes*

*Passeport Pour Unicode*

*LaTeX : apprentissage, guide et référence*

Une correction rapide du premier script (il y a bien sur plein de possibilité):

```
#!/bin/bash
```

```
PASTROUVE=1
```

```
affiche () {
  echo $2
  cat $1 | while read ligne
  do
    A=`echo ${ligne} | grep -i $2`
    if [ $? -eq 0 ]
    then
      PASTROUVE=0
      echo "${ligne}/*@"
    fi
  done
```



## ING1 : Correction Examen de Système d'Exploitation

Rédigé par : équipe pédagogique du cours de Syst. D'expl.

Ref : *ING1-SE-EXAMEN*

A l'intention de : Etudiants d'ING1

Créé le : 17/01/2012

#BONUS

```
if [ ${PASTROUVE} -eq 1 ]
then
  echo "pas de livre trouvé"
fi
}
```

#### Question 2

```
traitearg () {
  while [ $# -ne 0 ]
  do
    case $1 in
      "-h" ) usage ;;
      *) ;;
    esac
    shift
  done
}
```

```
usage () {
  echo "bla bla"
  exit 1
}
```

traitearg \$\*

#Question 3

```
if [ -f $1 -a -r $1 ]
then
  affiche $1 $2
else
  echo "fichier non accessible"
fi
```

Rédigé par : équipe pédagogique du cours de Syst. D'expl.

Ref : *ING1-SE-EXAMEN*

A l'intention de : Etudiants d'ING1

Créé le : 17/01/2012

## 2. Script shell (3 pts)

Écrire un script shell qui affiche un paramètre sur 3 de la ligne de commande.

```
#!/bin/bash
```

```
# Tant qu'il y a des arguments à traiter
```

```
while [ $# -gt 0 ]
```

```
do
```

```
  # j'affiche le premier argument
```

```
    echo $1
```

```
    # Puis j'en passe deux
```

```
    shift
```

```
    shift
```

```
    shift
```

```
done
```

## 3. Système de fichiers (3 pts)

L'indexation de bloc mémoire dans un système de fichier EXT2 est définie de cette manière :\*

- Les 10 premiers champs pointent chacun sur 1 bloc de données ;
- Le champ 11 (simple indirection) pointe vers  $256^1$  blocs de données ;
- Le champ 12 (double indirection) pointe vers  $256^2$  blocs de données ;
- Le champ 13 (triple indirection) pointe vers  $256^3$  blocs de données.

**1. Comment peut-on calculer la taille maximale d'un fichier dans ce système de fichier EXT2 ?**

$NBBlocs = (10 + 256 + 256^2 + 256^3) = 1678426610$

Taille = Taille Bloc \* NbBlocs

**2. Pour une taille d'un bloc de 1kB, quelle est cette taille maximale**

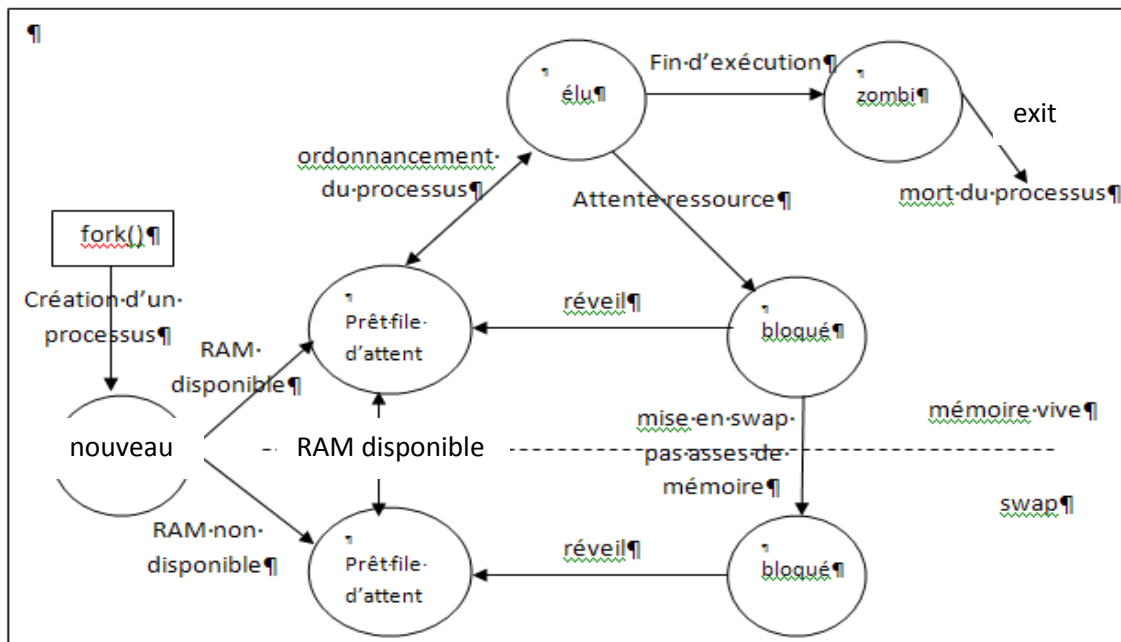
Taille = Taille Bloc \* NbBlocs =  $1024 * NbBlocs = 1024 * (10 + 256^1 + 256^2 + 256^3) = 17247250432$

La taille maximale d'un fichier avec le système de fichier EXT2 (en considérant les suppositions ci-dessus quant à la taille des blocs) est de 17 247 250 432 octets, soit environ 16 Gio (ou 17 Go).

## 4. Processus (5 pts)

Soit le schéma ci-dessous, les cercles représentent les différents états d'un processus

1. Donner le nom des 2 états manquants (voir le schéma)
2. En une phrase expliquer pour chacun de ces états comment et pourquoi un processus s'y trouve
3. Comment sont reliés ces états entre eux (sens des flèches), qualifier chaque relation (nom)
4. Remplacer le point d'interrogation par le nom de la fonction C sous linux la primitive fork()



**nouveau** : vient tout juste d'être créée par la primitive fork

**prêt** : Ce processus est mis dans la file d'attente en RAM S'il y a de la place sinon dans la mémoire de swap dédiée

**élu** : l'ordonnanceur lui attribue les ressources de calcul pendant un laps de temps et le remet dans la file d'attente

**bloqué** (ou endormi) : si le processus élu est en manque de données par exemple, il est mis dans la file de processus bloqués par la primitive sleep

Lorsque les données arrivent le processus est réveillé (wake up) et devient prêt s'il y a de la RAM de disponible, sinon il reste en état endormi dans le swap jusqu'à la libération de la mémoire.

**zombi** : le processus a terminé l'exécution de son programme, il devient zombi

Rédigé par : équipe pédagogique du cours de Syst. D'expl.

Ref : *ING1-SE-EXAMEN*

A l'intention de : Etudiants d'ING1

Créé le : 17/01/2012

### 5. Gestion de la mémoire (4 pts)

#### 1. Définir ce que c'est un défaut de page

Pour étendre l'espace d'adressage, indépendant de celui de la mémoire physique, on utilise le mécanisme de mémoire virtuelle

Si l'adresse virtuelle référence une page qui n'est pas présente en mémoire centrale alors le mécanisme d'adressage génère un **défaut de page**.

Si la mémoire physique RAM est pleine

- virer de la mémoire physique une page (**remplacement**) :
  - choisir une page "victime",
  - si elle a été modifiée, la réécrire sur disque,

#### 2. Où stocker les pages délogées de la RAM ?

On stocke sur un ou plusieurs disques durs, sur une zone appelée zone d'échange ou swap)

Le **swap** est traditionnellement une partition dédiée de taille fixe.

Mais le déchargement d'une page ne s'effectue pas toujours en swap, parfois la page n'est pas transférée mais juste déchargé et elle est à charge du système de la recharger si il y en a de nouveau besoin.

#### 3. Expliquer ce que c'est la partition de swap et le fichier de swap et donner les avantages et les inconvénients de chacun

Sur un disque local on peut trouver

- une partition de swap : plus rapide mais moins de place pour les systèmes de fichier
- fichier de swap : moins rapide mais plus de place pour les autres fichiers

En général, le système d'exploitation utilise les 2 simultanément

**Remarque** : Toutes les 3 questions de cours ont été traitées durant les présentations du cours en amphi. Je vous conseille d'être présents en amphi et surtout de prendre des notes. C'est valable pour tous les cours d'informatique