

Examen de l'UE "Introduction aux Systèmes et Réseaux" (FLIN302)

Michel Meynard

Durée : 1 heure; **Rédiger sur une copie séparée**

Tous documents autorisés

14 décembre 2009

1 Filtre

On désire écrire un programme C sous Unix `minMAJ.c` qui transforme tous les caractères minuscules (a-z) d'un fichier passé en paramètre en caractères majuscules correspondantes (A-Z). Les autres caractères (non minuscules) ne seront pas modifiés par ce programme.

Ce genre de fichier est **non borné** (il est donc impossible de le recopier en mémoire centrale dans un tableau) !

Par exemple :

```
>cat lafont.txt
Maitre Corbeau
sur 1 arbre perché
>minMAJ lafont.txt
MAITRE CORBEAU
SUR 1 ARBRE PERCHÉ
```

1.1 Questions

Répondre à ces questions en **2 à 3 phrases** claires et correctement écrites (orthographe et grammaire).

1. Donnez en hexadécimal la plage des codes ASCII des caractères minuscules.
2. Donnez en hexadécimal la plage des codes ASCII des caractères majuscules.
3. Pourquoi la lettre `é` n'est pas transformée par ce programme ?
4. Ecrire l'algorithme de ce programme.
5. Ecrire le programme `minMAJ.c`
6. On souhaite transformer ce programme en filtre Unix tel que `head` ou `grep`. Que faut-il faire pour lire l'entrée standard dans le cas où aucun paramètre fichier n'est fourni ? Ecrivez les modifications à effectuer.

L2 Informatique - Mathématiques
Examen "Introduction Système et Réseau" - FLIN302
Amphi Dumontet (126)
Tout document autorisé

Partie Pierre Pompidor

lundi 11 janvier 2009 - 13h-15h

Compréhension d'un script système écrit en Python : 10 points

Pour expliquer ce que fait le script suivant (examen.py), **décrivez les données sur lesquelles ce script va fonctionner, la façon dont il est appelé dans le terminal et son résultat.**

Par ailleurs, expliquez soigneusement le rôle des cinq expressions régulières utilisées.

Ce que je veux savoir, c'est ce que vous avez compris des fonctionnalités du script, et non pas que for est une instruction de boucle, ou que if est celle d'un test...

```
#!/usr/bin/python

import os, re, sys
resultats = {}

def parcours(r, e, f) :
    liste = os.listdir(r)
    for fichier in liste :
        res3 = re.search("^\\.\"", fichier)
        if not res3 :
            if os.path.isdir(r+"/"+fichier) :
                parcours(r+"/"+fichier, e, f)
            else :
                res4 = re.search("^"+f+"$", fichier)
                if res4 :
                    i = 0
                    fd = open(r+"/"+fichier)
                    if fd :
                        lignes = fd.readlines()
                        for ligne in lignes :
                            res5 = re.search(e, ligne)
                            if res5 : i += 1
                        fd.close()
                    if i > 0 :
                        if resultats.has_key(i) :
                            if resultats[i].has_key(r) :
                                resultats[i][r] += ":"+fichier
                            else :
                                resultats[i][r] = fichier
                        else :
                            resultats[i] = {}
                            resultats[i][r] = fichier
```

```

# Programme principal

if len(sys.argv) > 2 :
    exp = sys.argv[3]
    res1 = re.search("^*(.*)", exp)
    if res1 :
        exp = ".*"+res1.group(1)
    res2 = re.search("(.^[^\.])\*(.*)", exp)
    while res2 :
        exp = res2.group(1)+".*"+res2.group(2)
        res2 = re.search("(.^[^\.])\*(.*)", exp)
    parcours(sys.argv[1], sys.argv[2], exp)

clefs = resultats.keys()
clefs.sort()
for clef in clefs :
    print clef, "appariements :"
    for clef2 in resultats[clef].keys() :
        print "    ", clef2, ":", resultats[clef][clef2]

```