

Sistema Eragileak I - *Shell* programagarria

1. *auskalo* izena duen hurrengo scripta emanik:

```
for i in *
do
  if [ -d $i ]
  then
    for j in $i/*
    do
      if [ -x $j ]
      then
        echo $j
      fi
    done
  fi
done
```

- a) Zer egiten du *auskalo* scriptak?
- b) Sinplifikatu *auskalo* scripta, hurrengo komandoa erabiliz (zein_x)

zein_x katalogoa

Irteera estandarrean katalogoaren barnean dauden eta exekuzio baimena duten sarrera guztiak idazten ditu.

2. Idatzi script bat uneko katalogoa aztertzen duena honakoa eginez: aurkitutako fitxategi bakoitzeko, kopia egiten du /usr katalogoan, eta aurkitutako katalogo bakoitzeko, bere izena idazten du irteera estandarrean.

3. *auskalo* izena duen hurrengo scripta emanik:

```
for i in *
do
  if [ -d $i ]
  then
    for j in $*
    do
      if [ -f $i/$j ]
      then
        echo $i/$j >> ema
        head $i/$j >> ema
      fi
    done
  fi
done
more ema
rm ema
```

- a) Zer lortzen da hurrengo exekutatu ondoren? *auskalo* html gif jpg
- b) Zer egiten du *auskalo* scriptak?
- c) Aldatu *auskalo* ezaba ditzan parametro bezala pasatuko amaiera duten fitxategi guztiak, uneko katalogoaren azpikatalogo guztietatik. Adibidez: ezabatu gif jpg scriptak, “gif” edo “jpg” amaiera duten fitxategi guztiak ezabatu behar ditu uneko katalogoaren azpikatalogo guztietatik.

4. Hurrengo aplikazioa eraiki nahi da, *hustu_kat*, zeinak katalogo baten edukia ezabatzen duen, baina aukera eman ez eduki guztia berreskuratzeko.

Sistema Eragileak I - *Shell* programagarria

5. *auskalo* izena duen hurrengo scripta emanik:

```
for i in *
do
  if [ -d $i ]
  then
    for j in $i/*
    do
      if [ -f $j ]
      then
        echo $j
      fi
    done
  fi
done
```

- a) Zer egiten du *auskalo* scriptak?
b) Aldatu *auskalo* scripta hurrengoan egin dezan:
auskalo kat1 fitx2
kat1 katalogora mugitzen ditu Uneko katalogoaren azpikatalogoetan dauden eta fitx2 amaiera duten fitxategiak, kat1 katalogora mugituz.

Adibidez:
auskalo /tmp .o

Uneko katalogoaren azpikatalogoetan dauden fitxategi guztiak aztertu, eta ".o" amaiera duten fitxategiak /tmp katalogora mugitzen ditu.

6. Ondoko UNIX komandoak erabiliz:

- ls kat** Irteera estandarrean parametro bezala pasatako katalogoaren edukia (fitxategiak eta azpikatalogoak) idazten du. Parametrorik pasatzen ez bazaio, uneko katalogoarekin burutzen du.
- cat fitx1 fitx2 ...** Parametro bezala pasatako fitxategiak, bata bestearen atzetik kateatzen ditu eta irteera estandarrean idazten du edukia.
- cp fitx1 fitx2** *fitx1* fitxategia *fitx2* fitxategian kopiatzen du, azken hau sortuz.
- grep hitza** Parametro bezala pasatako *hitza* hitza bilatzen du sarrera estandarrean, eta topatutako lerro bakoitza irteera estandarrean idazten du.
- sort** Sarrera estandarrean idatzitakoa, alfabetikoki lerroka ordenatzen du eta irteera estandarrean idazten du.
- nl** Sarrera estandarreko lerroak irteera estandarrean idazten ditu, aurretik zenbatgarren lerroa den adieraziz.

(A) Idatzi komando-lerro bat, *index* izeneko fitxategia sortzen duena, honako edukiarekin: uneko katalogoaren edukiaren zerrenda ordenatua alfabetikoki eta zenbakitua izango du, "*doc*" azpikatearekin hasten direnak soilik agertuko direlarik.

(B) Zer gertatzen da hurrengo *auskalo* scripta exekutatzera?

```
for i in t*.c
do
  echo ===== $i =====
  sort $i | nl | grep $1
done
```

Sistema Eragileak I - *Shell* programagarria

- (C) Script bat idatzi, uneko katalogoan, **kodea** izena duen azpikatalogoa sortu, eta parametro bezala pasatako amaierarekin bukatzen diren fitxategiak uneko katalogotik **kodea** katalogora kopiatuko ditu. Adibidez, parametro bezala "c" amaiera pasatzen bazaio, ".c" amaiera duten fitxategiak **kodea** katalogora kopiatu beharko dira.

7. **auskalo** izena duen hurrengo scripta emanik:

```
if [ $# -ne 1 ]
then
    echo "errore"
    exit 1
fi
mkdir /backup/$1
for i in *.$1
do
    if [ -f $i ]
    then
        cp $i /backup/$1/$i
    fi
done
```

- b) Aldatu **auskalo** scripta, lehenik eta behin /tmp/zaborra katalogoa sortzeko, eta ondoren karpeta honetara mugitzea uneko katalogoaren azpikatalogoetan dauden fitxategiak, zeintzuk parametro bezala pasatako amaieraren baten berdina dutenak. Aztertu behar da gutxienez parametro bat pasatzen dela. Adibidez:

```
auskalo bmp gif jpg
```

Uneko katalogoan 3 azpikatalogo badaude (*kat1*, *kat2* eta *kat3*), azpikatalogo hauetan dauden eta ".bmp", ".gif" edo ".jpg" amaiera duten fitxategiak /tmp/zaborra katalogora mugitu behar dira.

- a) Zer egiten du **auskalo** scriptak?

8. **auskalo** izena duen hurrengo scripta emanik:

```
if [ $# -ne 1 ]
then
    echo "errore"
    exit 1
fi
for i in *
do
    if [ -d $i ]
    then
        for j in $i/$1*
        do
            if [ -x $j ]
            then
                echo "Aurkitua:" $j
            fi
        done
    fi
done
```

- a) Zer egiten du **auskalo** scriptak?
- b) Hainbat script programatu dira, guztiak ".sh" amaierarekin, eta uneko katalogoaren azpikatalogo guztietan sakabanatuta gorde dira. Aldatu **auskalo** scripta hurrengo egin dezan: bilatu azpikatalogoetan script guztiak, eta bakoitzeko, begiratu ea exekuzio baimena duen, eta izanez gero mezu bat idatzi adieraziz exekuta daitekeela; aldez, ez badu exekuzio baimenik, exekuzio baimena ipini beharko zaio eta ondoren mezu bat idatzi, esanez baimena aldatua izan dela.

9. *auskako* izena duen hurrengo scripta emanik:

```

for i $*
do
    if [ -d $i ]
    then
        for j in $i/*
        do
            if [ -x $j ]
            then
                echo $j
            fi
        done
    fi
done

```

a) Zer egiten du *auskalo* scriptak?

b) ***errepikatuak_bilatu*** scripta programatu ondokoa egin dezan: uneko katalogoko fitxategi motako sarrera bakoitza lehen mailako azpikatalogo bakoitzean bilatuko du, eta aurkitzen badu, bere izena uneko katalogoko ***errepikatuak.txt*** fitxategiari gehituko dio.

10. Ondoko *auskalo* izeneko scripta emanik, azal ezazu zer egiten duen.

```

for i in A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
do
    mkdir /$i
    for j in $i*
    do
        if [ -f $j ]
        then
            cp $j /$i
        else if [ -d $j ]
        then
            for k in $j/$i*
            do
                if [ -f $k ]
                then
                    cp $k /$i
                fi
            done
        fi
    done
done
done

```

b) ***find_files*** scripta programatu ondokoa egin dezan: Bi argumentu jasoko ditu, lehena fitxategi baten izena, eta bigarrena fitxategi amaiera bat. Uneko katalogoaren azpikatalogoetan (maila bat bakarrik jaitziz), bigarren argumentuarekin amaitzen diren fitxategien izenak lehen argumentu bezala pasatako fitxategian gehituko dira.

Adibidez: `find_files emaitza.txt .c`

emaitza.txt fitxategian, uneko katalogoaren azpikatalogoetan (maila bat jaitziz) ***.c*** amaiera duten fitxategi guztien izenak gehituko ditu.