

Programación Declarativa

2007-2008

Práctica 11

echo

Escriba un programa, *echo*, que repita por la salida estándar cada línea de texto que sea introducida por la entrada estándar.

```
Ej.:
echo
hola
hola
adiós
adiós
```

mcd

Escriba un programa, *mcd*, que calcule y visualice por la salida estándar el máximo común divisor de dos números naturales arbitrariamente grandes. Los dos números se indicarán como parámetros al invocar el programa desde la línea de comandos.

```
Ej.:
mcd 10 8
2
```

mcm

Escriba un programa, *mcm*, que calcule y visualice por la salida estándar el mínimo común múltiplo de dos números naturales arbitrariamente grandes. Los dos números se indicarán como parámetros al invocar el programa desde la línea de comandos.

```
Ej.:
mcm 10 8
40
```

factor

Escriba un programa, *factor*, que visualice por la salida estándar los factores primos de un número natural arbitrariamente grande (uno por línea y de menor a mayor). El número a factorial se indicará como parámetro al invocar el programa desde la línea de comandos

```
Ej.:
factor 24
2
2
2
3
```

nextprime

Escriba un programa, *nextprime*, que halle y visualice por la salida estándar el primer número primo mayor que un número natural dado arbitrariamente grande, que se indicará como parámetro al invocar el

programa desde la línea de comandos

```
Ej.:
nextprime 8
11
```

sort

Escriba un programa, *sort*, que visualice por la salida estándar las líneas de un archivo de texto, en orden alfabético. El nombre del archivo cuyas líneas hay que visualizar se indicará como parámetro al invocar el programa desde la línea de comandos.

```
Ej.:
sort archivo.txt
ab
abcd
abdc
Primera línea
Última línea
```

finddup

Escriba un programa, *finddup*, que busque las líneas comunes en dos archivos de texto. Los nombres de los archivos de texto se indicarán como parámetros al invocar el programa desde la línea de comandos. El programa deberá mostrar por la salida estándar cada una de las líneas que aparezca en ambos archivos, precedida de las posiciones (número de línea) que ocupan en cada uno de ellos. Se escribirá primero la posición en la que aparece la línea en el primer archivo y luego (separada sólo con una coma y sin espacios en blanco) la posición en la que aparece la línea en el segundo archivo. A continuación se escribirá el carácter "dos puntos" e inmediatamente la línea común.

```
Ej.:
finddup ar1.txt ar2.txt
2,10:una línea común
12,12:otra línea común
```

permut

Escriba un programa, *permut*, que muestre por la salida estándar todas las permutaciones posibles de una cadena de caracteres dada. La cadena de caracteres (que estará formada sólo por letras del alfabeto y no contendrá espacios en blanco) se le indicará al programa como parámetro al invocarlo desde la línea de comandos. Las permutaciones se mostrarán una por línea y por orden alfabético (sin repeticiones).

```
Ej.:
permut aba
aab
aba
baa
```