

# Pràctica Estructures de la informació

## Codificador - Final

Ja som capaç de codificar i/o decodificar!

Biel Moyà

2013-2014

Amb els fitxers *\*.de* i *\*.co* ja tenim tot allò necessari. Ara només falta realitzar un segon programa que ens permeti fer les feines de codificar i/o decodificar una entrada.

Aquest programa serà cridat de la següent manera

`nomprograma lletra arxiu1 arxiu2`

on

- **nomprograma:** és el nom del programa.
- **lletra:** ens servirà com a opció per a codificar(c) / decodificar (d).
- **arxiu1:** nom de l'arxiu d'on llegirem el que volem codificar / decodificar.
- **arxiu2:** nom de l'arxiu que conté la informació necessària per a realitzar la feina.

La sortida d'aquest programa serà per pantalla i a un nou arxiu.

## Processos

Els dos processos, no son més que un senzill tractament sequencial de caràcters.

## Codificació

El procés de codificació és molt senzill:

- carregar taula de codis
- mentre no final de fitxer

- llegir caràcter de l'*arxiu1*
- imprimir (sigui per pantalla, sigui a un fitxer) el còdi corresponent de la taula

## Decodificació

El procés de la decodificació necessita una mica més de feina. En primer lloc, en la fase anterior **necessitam haver guardat en un fitxer de text** el nombre de estats que hem generat (\*.n).

El procés és el següent:

- cal dimensionar les matrius de l'autòmat i carregar-les amb la informació de les transicions
- estat actual = 1
- mentre no final de fitxer
  - llegir símbol (0/1)
  - estat actual = consultar taula
  - Si l'estat següent és final
    - \* consultar taula caràcters
    - \* imprimir símbol.
    - \* estat actual = 1

## Llegir fitxers d'enters

Per tal de simplificar la feina, aquí teniu un exemple de com llegir un fitxer d'enters:

```
with ada.text_io,ada.integer_Text_io, Ada.Sequential_IO;
use ada.text_io;
procedure main is

f : File_type;
i : integer;
c:character;
begin
    Open(f,In_File,"text");
    while not end_of_File(f) loop
        get(f,c);
        i := character'pos(c) - 48;
        ada.integer_text_io.put(i);
    end loop;
```

```
        close(f);  
end main;
```