



UNIUNEA EUROPEANĂ



GUVERNUL ROMÂNIEI



Instrumente Structurale  
2007-2013



# Platformă de e-learning și curriculum e-content pentru învățământul superior tehnic

## Transmisia datelor multimedia in rețele de calculatoare

### 16. Algoritmul Lempel-Ziv

# Algoritmul Lempel-Ziv

- Urmatorul exemplu se refera la o versiune simpla a algoritmului Lempel-Ziv
- Schema de compresie trebuie sa imparta datele in sub-siruri, sa codeze sub-sirurile, si sa fie posibila si operatia inversa
- Fie urmatorul sir de date:

001101100011010101001001001101000001010010110010  
110

# Algoritmul Lempel-Ziv

- **Primul pas** consta in impartirea sirului de date in subsiruri, astfel incat la fiecare definitie a unui subsir sa nu existe repetitii, deci el sa nu fi fost definit anterior
  - Se va utiliza virgula ca separator de sub-siruri
  - La inceputul sirului se pune o virgula pentru a evidentia sirul de lungime zero
  - Se pune apoi o virgula dupa primul zero, intrucat nu a mai aparut
  - Al doilea bit este zero si se considera si al doilea simbol, 1, obtinandu-se sirul 01
  - Intrucat acesta este sir nou se pune virgula
  - Urmeaza simbolul 1, care fiind caracter nou, atrage virgula dupa el
  - Apoi apare un zero, mai trebuie un zero (deci inca un simbol), ca sa fie un sir nou
  - Rezultatul consta in urmatoarea lista de siruri

,0,01,1,011,00,0110,10,101,001,0010,  
01101,000,00101,001011,0010110

# Algoritmul Lempel-Ziv

- **Pasul doi** consta in derminarea numarului de biti pentru reprezentarea secventei astfel obtinute
- Practic, se numeroteaza sirurile incepand cu primul sir de lungime nenula
- Se determina numarul de biti dupa relatia

$$k = \lceil \log_2 N \rceil$$

unde  $N$  reprezinta numarul de siruri si paranteza are rolul de rontunjire la cel mai mic intreg

- Pentru primul exemplu considerat se constata ca sunt 16 simboluri (inclusiv sirul de lungime zero) si sunt necesari 4 biti pentru reprezentarea fiecarui sir

# Algoritmul Lempel-Ziv

- **Pasul trei** consta in codarea sirurilor, astfel obtinute
- Se completeaza un tabel de forma de mai jos, in care se definesc:
  - *sirul*, numarul ce arata pozitia sirului
  - *prefixul*, numarul ce arata pozitia prefixului
  - *codul* sirului
- Codul sirului este obtinut considerand numarul ce arata pozitia prefixului urmat de ultimul bit al sirului considerat, asa cum se prezinta in tabelul de mai jos

# Algoritmul Lempel-Ziv

| Nr. sirului | Sirul   | Codarea<br>pozitiei<br>sirului | Prefix | Codarea<br>pozitiei<br>prefixului | Sir codat      |
|-------------|---------|--------------------------------|--------|-----------------------------------|----------------|
| 1           | 0       | 0001                           | empty  | 0 = 0000                          | 0000+0 = 00000 |
| 2           | 01      | 0010                           | 0      | 1 = 0001                          | 0001+1 = 00011 |
| 3           | 1       | 0011                           | empty  | 0 = 0000                          | 0000+1 = 00001 |
| 4           | 011     | 0100                           | 01     | 2 = 0010                          | 00101          |
| 5           | 00      | 0101                           | 0      | 1 = 0001                          | 00010          |
| 6           | 0110    | 0110                           | 011    | 4 = 0100                          | 01000          |
| 7           | 10      | 0111                           | 1      | 3 = 0011                          | 00110          |
| 8           | 101     | 1000                           | 10     | 7 = 0111                          | 01111          |
| 9           | 001     | 1001                           | 00     | 5 = 0101                          | 01011          |
| 10          | 0010    | 1010                           | 001    | 9 = 1001                          | 10010          |
| 11          | 01101   | 1011                           | 0110   | 6 = 0110                          | 01101          |
| 12          | 000     | 1100                           | 00     | 5 = 0101                          | 01010          |
| 13          | 00101   | 1101                           | 0010   | 10 = 1010                         | 10101          |
| 14          | 001011  | 1110                           | 00101  | 13 = 1101                         | 11011          |
| 15          | 0010110 | 1111                           | 001011 | 14 = 1110                         | 11100          |

# Algoritmul Lempel-Ziv

- Secventa comprimata este format prin concatenarea tuturor sirurilor codate, aflate in ultima coloana a tabelului de mai sus
- Se obtine:

00000-00011-00001-00101-00010-01000-00110-01111  
01011-10010-01001-01010-10101-11011-11100

- Comparand lungimea secventei comprimate cu lungimea secventei originale se constata ca secventa comprimata are o lungime mai mare
  - Acest rezultat este explicat prin faptul ca secventa de intrare este de lungime foarte mica
  - Pentru fisiere cu secventa de lungime de milioane de simboluri se constata cu raport de compresie de pana la 2, depinzand si de continutul fisierului