

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Τεχνολογία Πολυμέσων

Ενότητα # 9: Κωδικοποίηση εντροπίας

Διδάσκων: Γεώργιος Ξυλωμένος

Τμήμα: Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Οικονομικό Πανεπιστήμιο Αθηνών**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



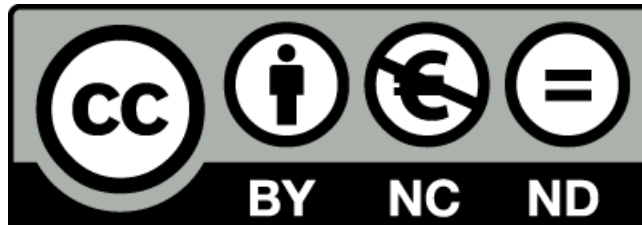
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Οι εικόνες προέρχονται από το βιβλίο «Τεχνολογία Πολυμέσων και Πολυμεσικές Επικοινωνίες», Γ.Β. Ξυλωμένος, Γ.Κ. Πολύζος, 1<sup>η</sup> έκδοση, 2009, Εκδόσεις Κλειδάριθμος.



# Σκοποί ενότητας

- Κατανόηση των βασικών τεχνικών κωδικοποίησης εντροπίας με χρήση δένδρου (Shannon-Fano και Huffman).
- Εξοικείωση με την τεχνική της αριθμητικής κωδικοποίησης.
- Εισαγωγή στις τεχνικές κωδικοποίησης με παράθυρο (LZ77/LZSS) και με λεξικό (LZ78/LZW).

# Περιεχόμενα ενότητας

- Κωδικοποίηση Shannon-Fano
- Κωδικοποίηση Huffman
- Αριθμητική κωδικοποίηση
- Κωδικοποίηση με παράθυρο
- Κωδικοποίηση με λεξικό

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**

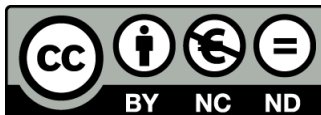


**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Κωδικοποίηση Shannon-Fano

**Μάθημα:** Τεχνολογία Πολυμέσων, **Ενότητα # 9:** Κωδικοποίηση εντροπίας

**Διδάσκων:** Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Βέλτιστη κωδικοποίηση

- Βέλτιστη κωδικοποίηση εντροπίας
  - bits/σύμβολο = εσωτερική πληροφορία
  - Απαιτούνται κωδικοί μεταβλητού μήκους
    - Πρόβλημα για την αποκωδικοποίηση
    - Πώς ξέρουμε που τελειώνει ένας κωδικός
  - Απαιτείται γνώση των πιθανοτήτων εισόδου
    - Είτε υποθέτουμε κάποια κατανομή
    - Είτε διαβάζουμε πρώτα όλο το αρχείο

# Μέθοδος Shannon-Fano (1 από 5)

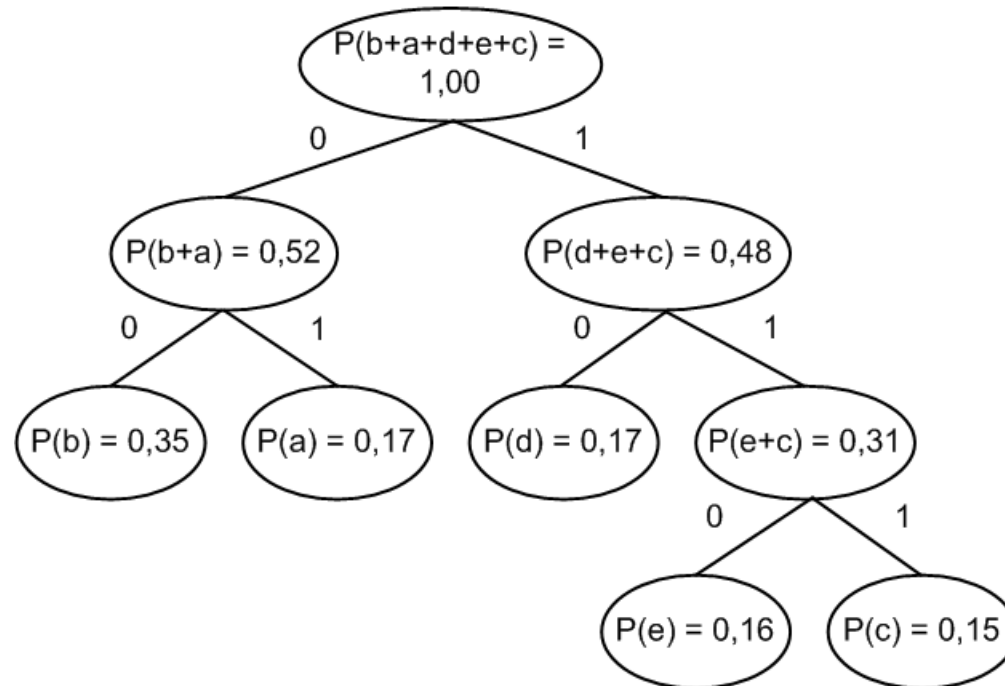
- Κωδικοποίηση Shannon-Fano
  - Χρήση κωδικών με ακέραιο μήκος bit
    - Απόκλιση από ιδανική συμπίεση
  - Δυαδικό δένδρο κωδικοποίησης
    - Φύλλα: σύμβολα και πιθανότητες
    - Κόμβοι: σύνολα συμβόλων και πιθανοτήτων
  - Ίδιο δένδρο αποκωδικοποίησης
    - Κάθε κωδικός έχει διαφορετικό πρόθεμα
    - Επιτρέπει αποκωδικοποίηση μεταβλητού μήκους



# Μέθοδος Shannon-Fano (2 από 5)

- Κατασκευή δένδρου
  - Ταξινομούμε σύμφωνα με τις πιθανότητες
  - Διάσπαση συμβόλων σε «ισοπίθανες» ομάδες
    - Κάθε ομάδα έχει το άθροισμα των πιθανοτήτων
    - Ομάδες με ελάχιστη διαφορά αθροίσματος
    - Οι δύο ομάδες γίνονται παιδιά του κόμβου
    - Αντιστοίχιση 0 και 1 στα δύο παιδιά
  - Επανάληψη μέχρι να μείνουν μόνο τα φύλλα
    - Κάθε φύλλο είναι ένα σύμβολο

# Μέθοδος Shannon-Fano (3 από 5)



- Παράδειγμα δένδρου κωδικοποίησης
  - Η ταξινόμηση δίνει τη σειρά b, a, d, e, c
  - Μέσο μήκος κώδικα: 2,31

# Μέθοδος Shannon-Fano (4 από 5)

- Κωδικοποίηση: Αντικατάσταση  $x$  με  $w(x)$ 
  - Κάθε σύμβολο αντιστοιχεί σε ένα φύλλο
  - Το μονοπάτι του φύλλου είναι ο κωδικός
- Αποκωδικοποίηση
  - Απαιτείται γνώση δένδρου κωδικοποίησης
  - Αντιστοίχιση εισόδου με μονοπάτια δένδρου
    - Κάθε πρόθεμα αντιστοιχεί σε διαφορετικό μονοπάτι
    - Ξέρουμε πάντα πότε να σταματήσουμε

# Μέθοδος Shannon-Fano (5 από 5)

- Κατασκευή δένδρου κωδικοποίησης
  - Κωδικοποίηση δύο φάσεων
  - Χρήση έτοιμων δένδρων
    - Ουσιαστικά έτοιμων κατανομών πιθανότητας
- Κατασκευή δένδρου αποκωδικοποίησης
  - Μετάδοση δένδρου κωδικοποίησης
  - Μετάδοση πιθανοτήτων
    - Απαιτείται γνώση των κανόνων κωδικοποίησης

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Κωδικοποίηση Huffman

**Μάθημα:** Τεχνολογία Πολυμέσων, **Ενότητα # 9:** Κωδικοποίηση εντροπίας

**Διδάσκων:** Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



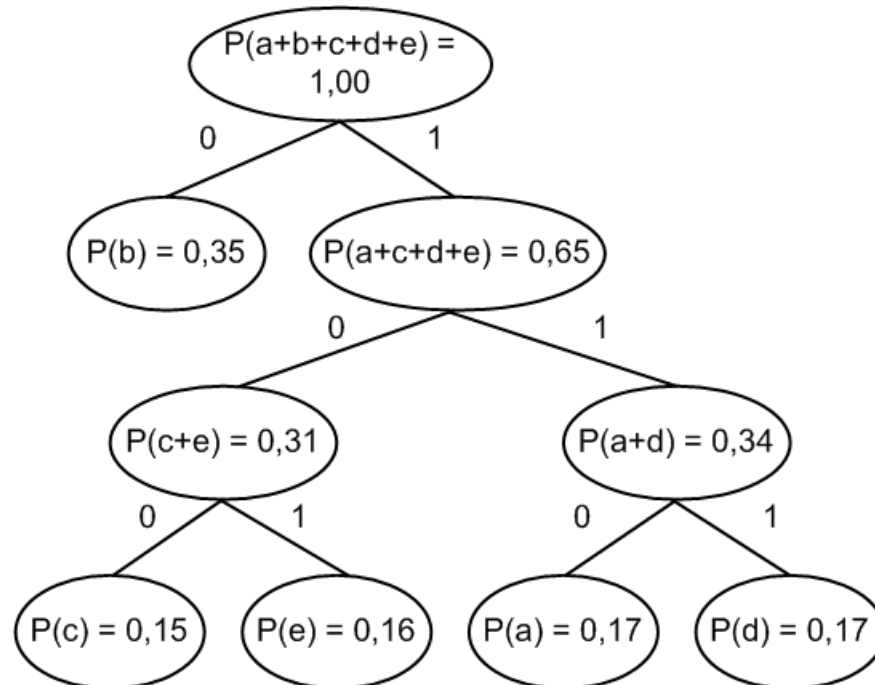
# Μέθοδος Huffman (1 από 5)

- Παρόμοια με Shannon-Fano
  - Μεταβλητό πλήθος bit ανά σύμβολο
  - Απαιτεί γνώση πιθανοτήτων εισόδου
  - Δυαδικό δένδρο (απο)κωδικοποίησης
    - Μπορεί να διαφέρει από της Huffman
  - Ίδιος αλγόριθμος (από)κωδικοποίησης
  - Δημιουργία δένδρου με ανάποδο τρόπο

# Μέθοδος Huffman (2 από 5)

- Κατασκευή
  - Επιλογή κόμβων με ελάχιστες πιθανότητες
  - Αντικατάσταση κόμβων με υποδένδρο
    - Άθροισμα πιθανοτήτων παιδιών στον πατέρα
    - Αντιστοίχιση 0 και 1 στα δύο παιδιά
  - Σταματάμε όταν μείνει ένα δένδρο
    - Η ρίζα αντιπροσωπεύει όλα τα σύμβολα
  - Πιο ισορροπημένα δένδρα από Shannon-Fano

# Μέθοδος Huffman (3 από 5)



- Παράδειγμα δένδρου κωδικοποίησης
  - $P(a)=0,17$ ,  $P(b)=0,35$ ,  $P(c)=0,15$ ,  $P(d)=0,17$ ,  $P(e)=0,16$
  - Μέσο μήκος κώδικα: 2,3 (καλύτερο από Shannon-Fano)



# Μέθοδος Huffman (4 από 5)

- Huffman ή Shannon-Fano;
  - Σχεδόν πανομοιότυποι αλγόριθμοι
  - Πιο απλή κατασκευή στο Shannon-Fano
    - Όχι ταξινόμηση συμβόλων σε κάθε βήμα
  - Πιο αποδοτική κωδικοποίηση στο Huffman
    - Σε κάθε βήμα απαιτείται ταξινόμηση υποδένδρων
    - Γρήγορη υλοποίηση με σωρό

# Μέθοδος Huffman (5 από 5)

- Μειονεκτήματα Huffman/Shannon-Fano
  - Απαιτείται γνώση των πιθανοτήτων εισόδου
  - Η κωδικοποίηση δεν είναι βέλτιστη
    - Ακέραιο πλήθος bit ανά σύμβολο
    - Αναγκαστικά αποκλίνουμε από το ιδανικό
  - Βελτίωση συμπίεσης
    - Κωδικοποίηση ομάδων από  $n$  σύμβολα
    - Εκθετική αύξηση του πλήθους των συμβόλων

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Αριθμητική κωδικοποίηση

**Μάθημα:** Τεχνολογία Πολυμέσων, **Ενότητα # 9:** Κωδικοποίηση εντροπίας

**Διδάσκων:** Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Αριθμητική (1 από 9)

- Κωδικοποίηση εισόδου με έναν αριθμό
  - Το πλήθος bit εξαρτάται από το μήκος
    - Μπορεί να είναι πάρα πολύ μεγάλο
  - Δεν απαιτείται σταθερό πλήθος bit/σύμβολο
    - Αποφεύγουμε την απόκλιση από το ιδανικό
  - Απαιτείται γνώση των πιθανοτήτων εισόδου
  - Χρησιμοποιείται τερματικό σύμβολο στο τέλος
    - Για να ξέρουμε ότι τελείωσε η αποκωδικοποίηση

# Αριθμητική (2 από 9)

- Προετοιμασία αλγορίθμου
  - Ταξινόμηση συμβόλων (συνήθως αλφαβητικά)
  - Στο  $x_i$  αντιστοιχίζουμε το διάστημα  $[a_i, b_i)$ 
    - Το διάστημα ορίζεται έτσι ώστε  $b_i - a_i = p(x_i)$
  - Παράδειγμα
    - $P(a) = 0.4$ ,  $P(b) = 0.3$ ,  $P(c) = 0.2$  και  $P(\$) = 0.1$
    - Διάστημα a:  $[0, 0.4)$ , Διάστημα b:  $[0.4, 0.7)$
    - Διάστημα c:  $[0.7, 0.9)$ , Διάστημα \$:  $[0.9, 1.0)$

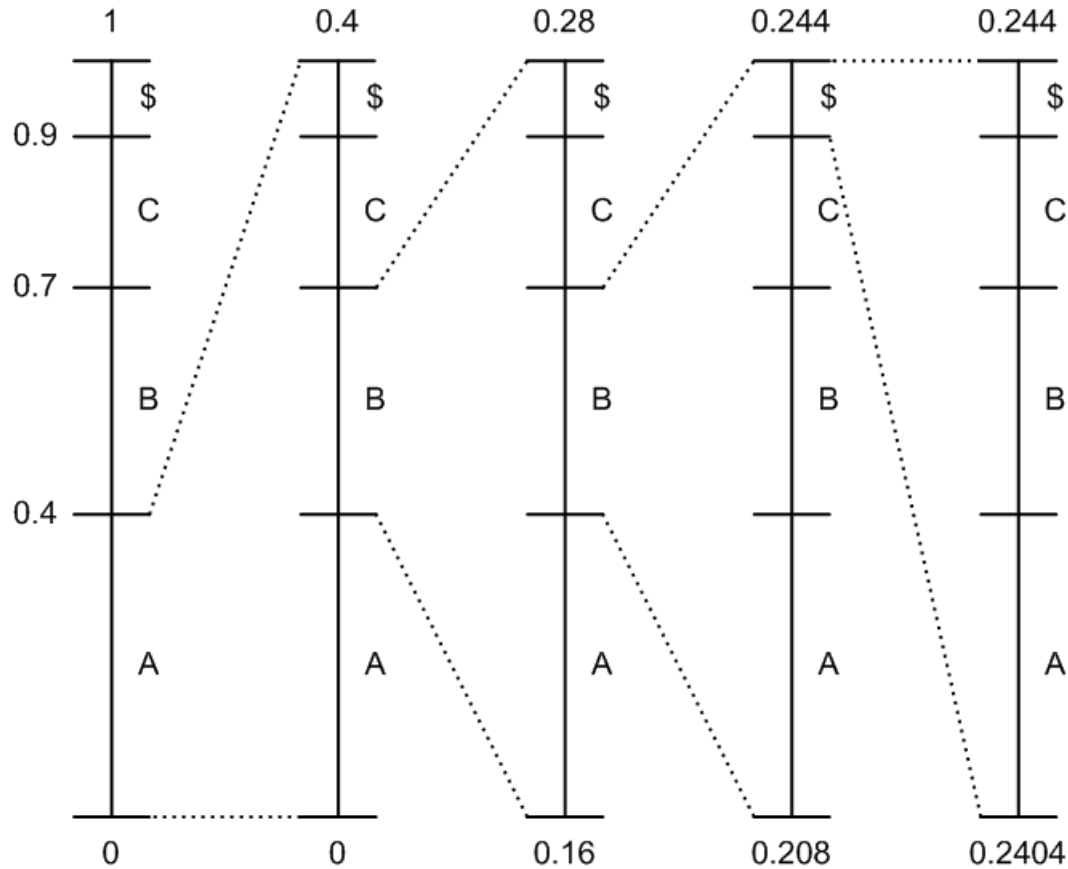
# Αριθμητική (3 από 9)

```
low = 0.0;
high = 1.0;
repeat {
    input s;
    range = high - low;
    high = low + range * highrange[s];
    low = low + range * lowrange[s];
} until s = $;
output any number in [low, high);
```

# Αριθμητική (4 από 9)

- Λειτουργία αλγορίθμου
  - Lowrange[]: κάτω όρια πιθανοτήτων
  - Highrange[]: άνω όρια πιθανοτήτων
  - Η είσοδος κωδικοποιείται με διάστημα
    - Αρχικά το διάστημα είναι  $[0,1)$
    - Σε κάθε βήμα το διάστημα περιορίζεται
    - Ανάλογα με το σύμβολο εισόδου
    - Όσο μεγαλύτερη είσοδος, τόσο μικρότερο διάστημα

# Αριθμητική (5 από 9)



- Παράδειγμα αριθμητικής κωδικοποίησης



# Αριθμητική (6 από 9)

- Υπολογισμός εξόδου
  - Θέλουμε έναν αριθμό μέσα στο διάστημα
  - Αλλά με τα λιγότερα δυνατά ψηφία!
  - Αρχίζουμε με 0, και προσθέτουμε κλασματικά
  - Επισυνάπτουμε μονάδα στο δεξί άκρο
    - Αν είμαστε πάνω από το άνω όριο, βάζουμε μηδέν
    - Αν είμαστε πάνω από το κάτω όριο, σταματάμε
  - Δεν χρειάζεται να στείλουμε το 0,

# Αριθμητική (7 από 9)

- Αλγόριθμος αποκωδικοποίησης

```
input n;
```

```
repeat {
```

```
    find s so that n is in  
    [lowrange[s], highrange[s]);
```

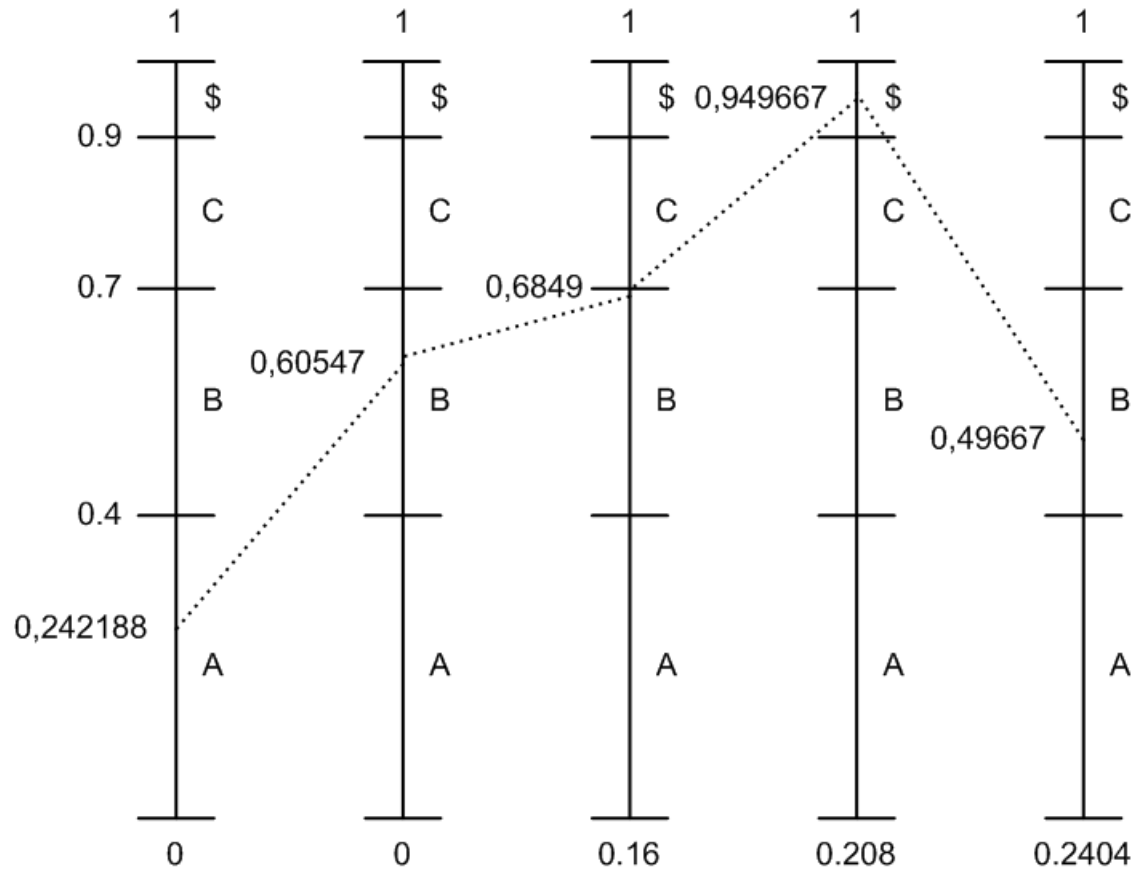
```
    output s;
```

```
    range = highrange[s] - lowrange[s];
```

```
    n = (n - lowrange[s]) / range;
```

```
} until s = $;
```

# Αριθμητική (8 από 9)



- Παράδειγμα αριθμητικής αποκωδικοποίησης

# Αριθμητική (9 από 9)

- Μειονεκτήματα αριθμητικής κωδικοποίησης
  - Διαχείριση αριθμών άγνωστου μήκους
    - Πάρα πολλά δεκαδικά ψηφία
    - Χρήση ειδικών βιβλιοθηκών
  - Κωδικοποίηση ομάδων σταθερού μήκους
    - Σπάμε την είσοδο σε μπλοκ
    - Κάθε μπλοκ απαιτεί μικρότερο αριθμό
    - Αλλά χάνουμε λίγο σε αποδοτικότητα

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**

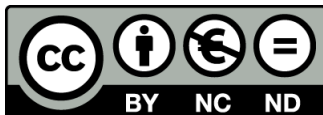


**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Κωδικοποίηση με παράθυρο

**Μάθημα:** Τεχνολογία Πολυμέσων, **Ενότητα # 9:** Κωδικοποίηση εντροπίας

**Διδάσκων:** Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο

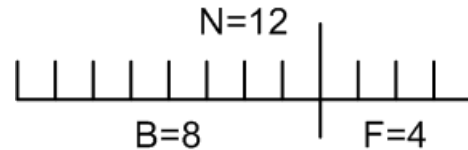


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Αλγόριθμος LZ77 (1 από 4)



- Αλγόριθμος LZ77 (Lempel-Ziv)
  - Σε κάθε στιγμή βλέπουμε παράθυρο της εισόδου
    - Αριστερό μέρος: κωδικοποιημένη είσοδος (λεξικό)
    - Δεξί μέρος: είσοδος προς κωδικοποίηση
  - Αντικατάσταση προθέματος με τριάδα (O,L,C)
    - O: θέση προθέματος στο αριστερό μέρος
    - L: μήκος του προθέματος που ταιριάζαμε
    - C: πρώτος χαρακτήρας που δεν ταιριάζει

# Αλγόριθμος LZ77 (2 από 4)

|a|c|a|b|b|a|c|a|b|a|a|c|

- Παράδειγμα κωδικοποίησης LZ77
  - Αντικαθιστούμε το baα με (4,2,a)
    - Θέση 4 στο παράθυρο
    - Μήκος ταιριάσματος 2
    - Επόμενο σύμβολο a
    - Η πρώτη θέση στο παράθυρο είναι 0
  - Αν δεν βρούμε ταιρίασμα, θέτουμε μήκος 0

# Αλγόριθμος LZ77 (3 από 4)

|a|c|a|b|b|a|c|a|a|a|c|b|

- Παράδειγμα με επικάλυψη
  - Επέκταση ταιριάσματος στο δεξί μέρος
  - Αντικαθιστούμε το aac με (7,2,c)
- Υλοποίηση κωδικοποίησης LZ77
  - Το παράθυρο συνήθως είναι δύναμη του 2
  - Παράδειγμα: 4096+4096 σύμβολα
  - Δείκτης: 12 bit για όλο το αριστερό μέρος
  - Μήκος: 12 bit για το μήκος του δεξιού μέρους



# Αλγόριθμος LZ77 (4 από 4)

- Εκκίνηση (απο)κωδικοποίησης
  - Υποθέτουμε γνωστό παράθυρο στα αριστερά
- Μειονεκτήματα LZ77
  - Κάθε τριάδα απαιτεί 4-5 byte στο παράδειγμα
  - Το αρχείο μεγαλώνει με κακά ταιριάσματα
  - Οι χαρακτήρες αρχικά κωδικοποιούνται ως (0,0,c)
    - Βελτίωση: αρχικό παράθυρο το σύνολο συμβόλων
  - Η κωδικοποίηση ξεκινά με παθητικό!

# Αλγόριθμος LZSS (1 από 2)

- Αλγόριθμος LZSS (Storer και Szymanski)
  - Παραλλαγή του LZ77
  - Διαφέρει στο τι παράγεται στην έξοδο
  - Δύο περιπτώσεις: ταίριασμα ή χαρακτήρας
  - Το πρώτο bit της εξόδου διακρίνει τι είναι
    - Είτε (O,L): θέση O, μήκος L
    - Είτε C: χαρακτήρας C χωρίς ταίριασμα
  - Η τριάδα σπάει σε δύο κομμάτια

# Αλγόριθμος LZSS (2 από 2)

- Υλοποίηση κωδικοποίησης LZSS
  - Δεν θέλουμε κωδικούς των 9 bit!
  - Χωρίζουμε την έξοδο σε ομάδες οκτώ κωδικών
  - Το πρώτο byte περιγράφει τους κωδικούς
    - Ένα bit για κάθε κωδικό
    - Δείχνει αν είναι ταίριασμα ή χαρακτήρας
    - Τα byte υπόλοιπα ερμηνεύονται αντίστοιχα
  - Διαβάζουμε συνέχεια ολόκληρα byte (ή λέξεις)

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Κωδικοποίηση με λεξικό

**Μάθημα:** Τεχνολογία Πολυμέσων, **Ενότητα # 9:** Κωδικοποίηση εντροπίας

**Διδάσκων:** Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Αλγόριθμος LZ78 (1 από 2)

- Δημιουργία και χρήση λεξικού
  - Δυνατότητα ταιριάσματος με παλιά είσοδο
  - Μέγιστο πρόθεμα εισόδου που υπάρχει στο λεξικό
  - Αντικατάσταση προθέματος με το ζεύγος (P,C)
    - P: θέση του προθέματος στο λεξικό
    - C: πρώτος χαρακτήρας που δεν ταιριάζει
  - Πρόθεμα + χαρακτήρας μπαίνουν στο λεξικό
  - Ο αποκωδικοποιητής χτίζει το ίδιο λεξικό
    - Χρήση του λεξικού για την αποκωδικοποίηση

# Αλγόριθμος LZ78 (2 από 2)

Είσοδος	Έξοδος	Λεξικό
a	(0,a)	Θέση 1: a
aa	(1,a)	Θέση 2: aa
b	(0,b)	Θέση 3: b
ba	(3,a)	Θέση 4: ba
ab	(1,b)	Θέση 5: ab

- Παράδειγμα: είσοδος aaabbaab
  - Το λεξικό σταδιακά αποκτά μεγάλες συμβολοσειρές
    - Στον LZ77 κάνουμε αμέσως μεγάλα ταιριάσματα
  - Ο αποκωδικοποιητής χτίζει το λεξικό από την έξοδο
    - Οι αναφορές γίνονται σε προηγούμενες καταχωρήσεις

# Αλγόριθμος LZW (1 από 10)

- Αλγόριθμος LZW (Welch)
  - LZ78: ακολουθεί τη λογική του LZ77
    - Σε κάθε ταίριασμα προσθέτει επόμενο χαρακτήρα
    - Εγγυημένη πρόοδος ακόμη και χωρίς ταίριασμα
  - Ο LZW παράγει μόνο δείκτες, όχι χαρακτήρες
    - Το λεξικό αρχικοποιείται με όλα τα σύμβολα
    - Άρα πάντα έχουμε κάποιο ταίριασμα
    - Οι συμβολοσειρές χτίζονται από αυτά τα σύμβολα

# Αλγόριθμος LZW (2 από 10)

- Κωδικοποίηση LZW
  - Μέγιστο πρόθεμα εισόδου που είναι στο λεξικό
  - Αντικαθίσταται από το δείκτη στο λεξικό
    - Ο επόμενος ανήκει στην επόμενη συμβολοσειρά
  - Προσθέτουμε πρόθεμα + επόμενο χαρακτήρα
    - Σταδιακή αύξηση μήκους συμβολοσειρών στο λεξικό
  - Η είσοδος προχωράει μετά το πρόθεμα
    - Ο επόμενος χαρακτήρας γίνεται αρχή της εισόδου



# Αλγόριθμος LZW (3 από 10)

```
input s;
while not EOF {
    input c;
    if [s+c] is in dictionary
        s = [s+c];
    else {
        output code(s);
        add [s,c] to dictionary with next code;
        s = c; }
}
output code(s);
```

# Αλγόριθμος LZW (4 από 10)

Είσοδος	Έξοδος	Λεξικό
		Θέση 1: a
		Θέση 2: b
a+a	1	Θέση 3: aa
aa+b	3	Θέση 4: aab
b+b	2	Θέση 5: bb
b+a	2	Θέση 6: ba
aab+b	4	Θέση 7: aabb

- Παράδειγμα: είσοδος aaabbaabb
  - Το λεξικό έχει αρχικά όλα τα σύμβολα

# Αλγόριθμος LZW (5 από 10)

- Αποκωδικοποίηση LZW
  - Διαβάζουμε τον κωδικό
  - Αν υπάρχει στο λεξικό, τον αντικαθιστούμε
    - Δεν εισάγουμε τρέχουσα συμβολοσειρά στο λεξικό
    - Δεν γνωρίζουμε τον επόμενο χαρακτήρα!
  - Εισάγουμε προηγούμενη + πρώτο χαρακτήρα
    - Τώρα μόνο γνωρίζουμε ποιος ήταν ο χαρακτήρας
    - Ο αποκωδικοποιητής είναι ένα βήμα πίσω

# Αλγόριθμος LZW (6 από 10)

- Αν ο κωδικός δεν υπάρχει στο λεξικό;
  - Έστω ότι η είσοδος είναι ο τελευταίος κωδικός
  - Δεν τον έχουμε προσθέσει ακόμη στο λεξικό!
    - Δεν γνωρίζουμε ακόμη τον επόμενο χαρακτήρα
  - Η συμβολοσειρά είναι της μορφής C???C
    - Μοναδικός τρόπος να συμβεί το παραπάνω
    - Άρα επαναλαμβάνουμε τον πρώτο χαρακτήρα
    - Αυτός αντιστοιχεί στον άγνωστο κωδικό

# Αλγόριθμος LZW (7 από 10)

```
s = NIL;
while not EOF {
    input c;
    entry = string(c);
    if entry not in dictionary
        entry = s + s[0];
    output entry;
    if (s != NIL)
        add [s,entry[0]] to dictionary with next code;
    s = entry;
}
```

# Αλγόριθμος LZW (8 από 10)

Είσοδος	Έξοδος	Λεξικό
		Θέση 1: a
		Θέση 2: b
1	a	
3	aa	Θέση 3: aa
2	b	Θέση 4: aab
2	b	Θέση 5: bb
4	aab	Θέση 6: ba

- Παράδειγμα αποκωδικοποίησης LZW
  - Ο κωδικός 3 δείχνει σε κενή θέση
  - Πρέπει να είναι παλιά + πρώτος χαρακτήρας (a+a)

# Αλγόριθμος LZW (9 από 10)

- Υλοποίηση λεξικού LZ78/LZW
  - Το λεξικό μεγαλώνει σε κάθε βήμα
  - Επεκτεινόμενοι δείκτες
    - Ξεκινάμε με δείκτες έστω 4 bit (16 θέσεις)
    - Όποτε γεμίζει το λεξικό, προσθέτουμε 1 bit
  - Τι γίνεται όταν το λεξικό μεγαλώσει πολύ;
    - Είτε σταματάμε να δεχόμαστε συμβολοσειρές
    - Είτε πετάμε τις λιγότερο χρησιμοποιημένες

# Αλγόριθμος LZW (10 από 10)

- Συμπύεση λεξικού LZ78/LZW
  - Κάθε νέα είσοδος επεκτείνει μία παλιά
    - Κατά έναν χαρακτήρα ακριβώς
  - Αποθήκευση δείκτη σε παλιά συμβολοσειρα
    - Και επιπλέον του νέου χαρακτήρα
  - Χρησιμοποιούνται ειδικές δομές δεδομένων
  - Επιτάχυνση ταιριάσματος και αντικατάστασης



**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Τέλος Ενότητας #9

**Μάθημα:** Τεχνολογία Πολυμέσων, **Ενότητα # 9:** Κωδικοποίηση εντροπίας

**Διδάσκων:** Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

