

# ΜΥΕ003: Ανάκτηση Πληροφορίας

*Διδάσκουσα: Ευαγγελία Πιτουρά*

Κεφάλαιο 1: Εισαγωγή. Ανάκτηση Boole

*Ακαδημαϊκό Έτος 2021-2022*

Αναζήτηση με λέξεις κλειδιά σε μεγάλες συλλογές κυρίως  
κειμένου

# Γιατί να μας ενδιαφέρει;

Παλιότερα,  
Βιβλιοθηκονόμους, βοηθούς νομικών  
επαγγελματιών κλπ;

**ISBN:** 0-201-12227-8

**Author:** Salton, Gerard

**Title:** Automatic text processing: the transformation, analysis,  
and retrieval of information by computer

**Editor:** Addison-Wesley

**Date:** 1989

**Content:** <Text>

**external attributes (metadata)** and **internal attribute** (content)

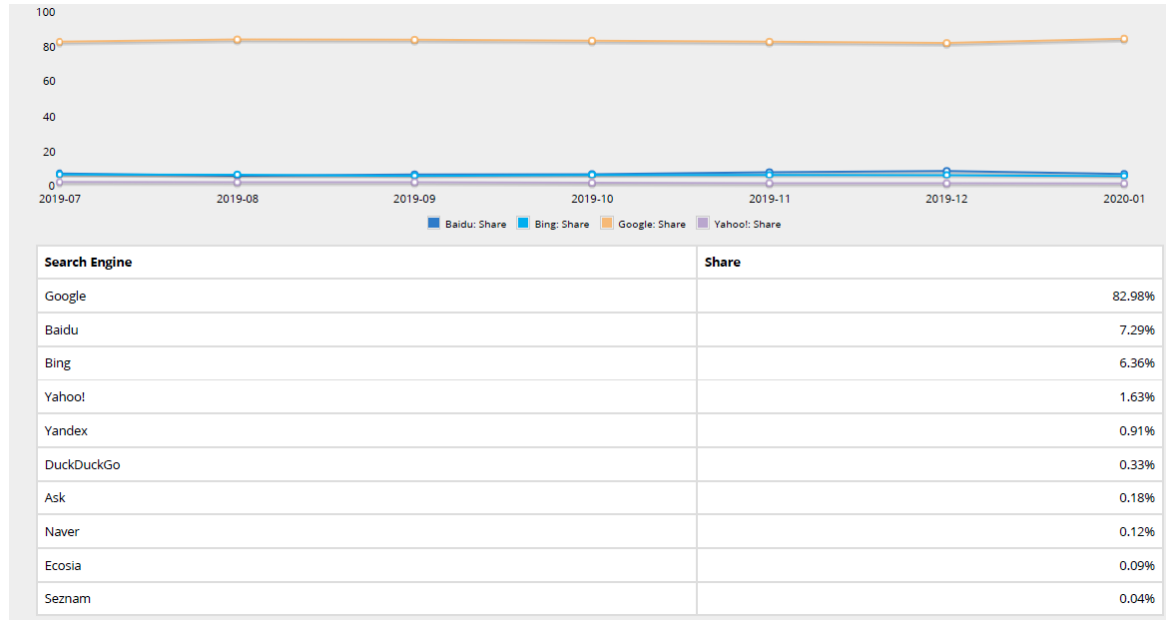
Search by external attributes = Search in DB

IR: search by content



# Εφαρμογές

## Web search



<https://www.netmarketshare.com/>, Jan 2020

**Baidu:** 2η μεγαλύτερη μηχανή αναζήτησης στον κόσμο, και κατέχει το 76.05% του μεριδίου αγοράς στην αγορά μηχανών αναζήτησης της Κίνας.

**Yandex:** ρωσική πολυεθνική τεχνολογική εταιρεία που ειδικεύεται σε υπηρεσίες και προϊόντα σχετικά με το Διαδίκτυο και θεωρείται η μεγαλύτερη εταιρεία τεχνολογίας της **Ρωσίας**. Λειτουργεί την μεγαλύτερη μηχανή αναζήτησης στη Ρωσία με μερίδιο αγοράς περίπου **65%**.

**DuckDuckGo** είναι διαδικτυακή μηχανή αναζήτησης που δίνει έμφαση στην **προστασία της ιδιωτικής ζωής** των χρηστών της και στην αποφυγή του “**φίλτρου φυσαλίδας**” των εξατομικευμένων αποτελεσμάτων αναζήτησης.

Πηγή: Wikipedia

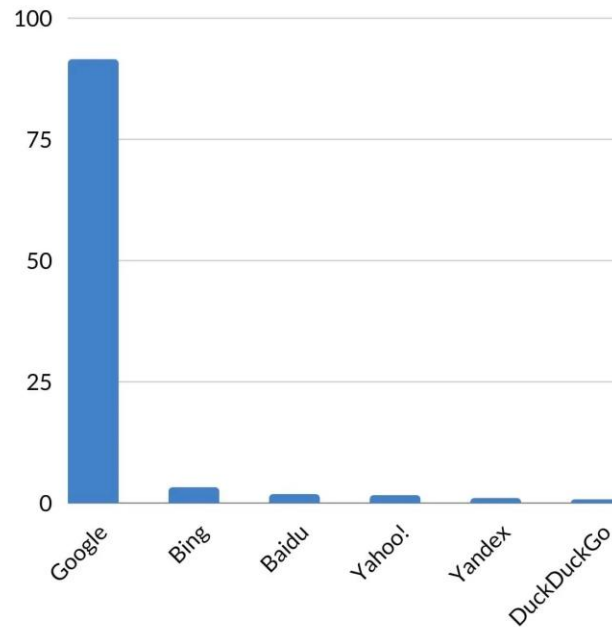
# Web search

## TOP SEARCH ENGINES

### GOOGLE

91.42%

Google is the most popular search engine with a stunning 91.42% market share compared to 3.14% of second in place Bing.



- Google
- Microsoft Bing
- Yahoo
- Baidu
- Yandex
- DuckDuckGo
- Ask.com
- Ecosia
- Aol.com
- Internet Archive

**Baidu:** 2η μεγαλύτερη μηχανή αναζήτησης στον κόσμο, και κατέχει το 76.05% του μεριδίου αγοράς στην αγορά μηχανών αναζήτησης της Κίνας.

**Yandex:** ρωσική πολυεθνική τεχνολογική εταιρεία που ειδικεύεται σε υπηρεσίες και προϊόντα σχετικά με το Διαδίκτυο και θεωρείται η μεγαλύτερη εταιρεία τεχνολογίας της **Ρωσίας**. Λειτουργεί την μεγαλύτερη μηχανή αναζήτησης στη Ρωσία με μερίδιο αγοράς περίπου **65%**.

**DuckDuckGo** είναι διαδικτυακή μηχανή αναζήτησης που δίνει έμφαση στην **προστασία της ιδιωτικής ζωής** των χρηστών της και στην αποφυγή του **“φίλτρου φυσαλίδας”** των εξατομικευμένων αποτελεσμάτων αναζήτησης.

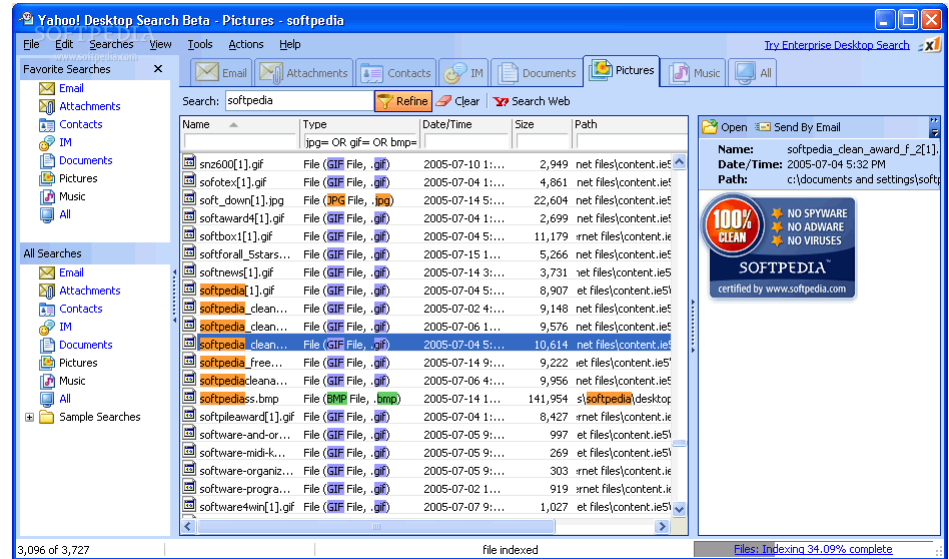
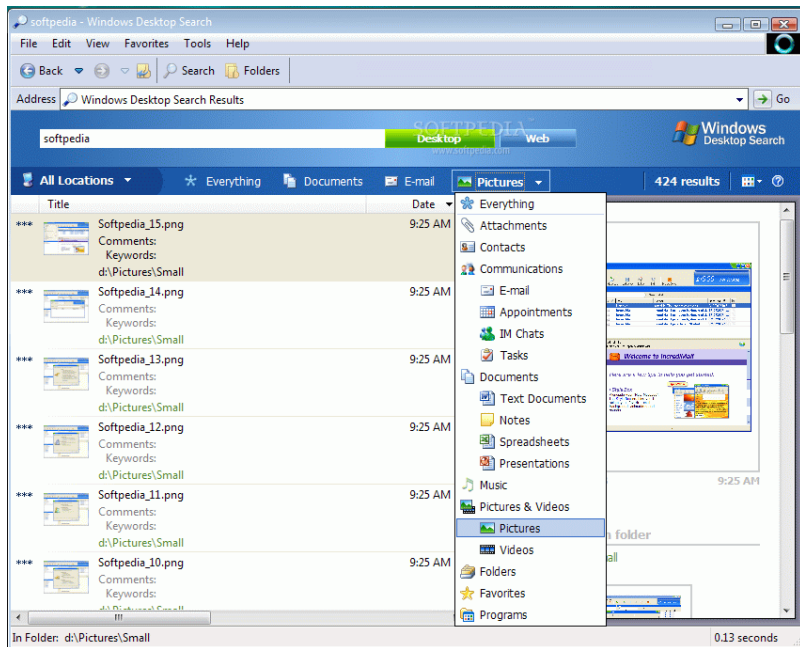
**Ask.com**

**Internet archive**

Πηγή: Wikipedia

<https://www.reliablesoft.net/top-10-search-engines-in-the-world/>

# Desktop search



# Εφαρμογές

Email search

Social search

Enterprise search

Domain specific search: Legal information  
retrieval, Digital libraries

Διαφορετικές απαιτήσεις ανάλογα με την εφαρμογή

# Κατηγορίες εφαρμογών

- Στο web/διαδίκτυο

Δισεκατομμύρια έγγραφα σε εκατομμύρια υπολογιστές.

Συλλογή εγγράφων, κλίμακα, διάταξη αποτελεσμάτων, ..

- Προσωπική ανάκτηση πληροφορίας

(στον προσωπικό υπολογιστή, email, κλπ)

Διαφορετικά είδη αρχείων, light-weight, maintenance-free, ...

- Σε επίπεδο επιχείρησης, οργανισμού (enterprise, institutional) και αναζήτηση ειδικού σκοπού (domain-specific search) – πχ ερευνητικά άρθρα σε βιοχημεία



# Ορισμός

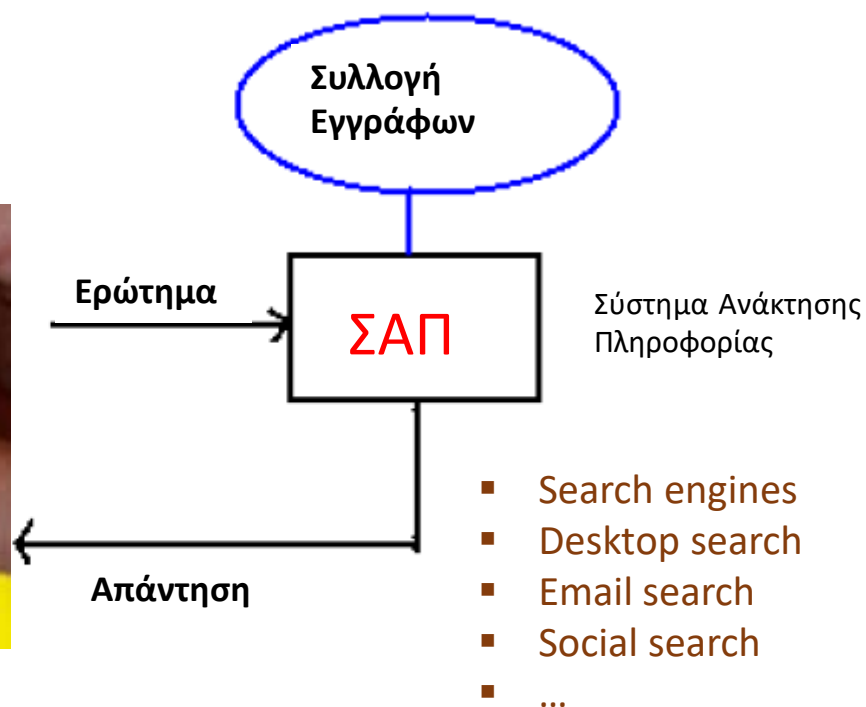
## Ανάκτηση Πληροφορίας (**Information Retrieval**) - (IR)

- είναι η εύρεση αντικειμένων κυρίως εγγράφων (**documents**) αδόμητης φύσης (\*) (**unstructured**) που συνήθως έχουν τη μορφή κειμένου (**text**)
- από μεγάλες συλλογές (συνήθως αποθηκευμένες σε υπολογιστές)
- τα οποία ικανοποιούν μια ανάγκη πληροφόρησης (**information need**)

*(\*) όχι ακριβώς!*

# Τι είναι η Ανάκτηση Πληροφορίας (Information Retrieval);

Ανάγκη  
πληροφόρησης



# Αδόμητα δεδομένα

- Τυπικά αναφέρεται σε *ελεύθερο κείμενο*
- Επιτρέπει
  - Ερωτήματα με **λέξεις κλειδιά** (keyword) με πιθανούς τελεστές
  - Ποιο περίπλοκες ερωτήσεις για **έννοιες**: π.χ.,
    - Βρες όλες τις web σελίδες για την απελευθέρωση των Ιωαννίνων
- Κλασσικό μοντέλο για αναζήτηση σε έγγραφα κειμένου

structured

	A	B	C	D	E	F	G
1	Purchase ID	Last name	First name	Birth day	Country	Date of purchase	Amount of purchase
2	1	Davidson	Michael	04/03/1986	United States	10/12/2016	37
3	2	Vito	Jim	09/01/1994	United Kingdom	02/02/2016	85
4	3	Johnson	Tom	23/08/1972	France	02/11/2016	83
5	4	Lewis	Peter	18/10/1979	Germany	22/11/2016	27
6	5	Koenig	Edward	13/05/1983	Argentina	26/03/2015	43
7	6	Preston	Jack	16/06/1991	United States	06/11/2016	77
8	7	Smith	David	11/03/1965	Canada	15/11/2016	23
9	8	Brown	Luis	03/09/1997	Australia	03/07/2015	74
10	9	Miller	Thomas	07/01/1980	Germany	07/11/2016	13
11	10	Williams	Bill	26/07/1960	United States	20/11/2015	80
12	11	Gemini	Alexia	12/09/1995	Canada	11/03/2017	35
13	12	Bond	James	25/02/1975	United Kingdom	12/08/2017	40
14	13	Burgle	Patricia	01/12/1990	United States	18/01/2015	55
15	14	Reding	Michelle	07/04/1985	Canada	23/02/2017	28
16	15	Harvey	Billy	14/07/1971	United Kingdom	12/01/2016	41
17							

unstructured

Introducing one of Australia's greatest treks and one of our newest trips! 🌄

The Cradle Mountain Overland track offers some of Tasmania's most stunning scenery – dramatic valleys, temperate rainforests, beautiful lakes and more. And this 6-day camping trip is the perfect way to experience it, alongside like-minded adventurers and experienced guides.

Why travel there with us? ... See More

View Similar Products

116

11 Comments 12 Shares

Most Relevant is selected, so some comments may have been filtered out.

Jessica Chapman  
\*sigh\* I miss travel. With COVID travel restrictions here it's going to be a while.  
Like · Reply · 1w  
↳ 1 Reply

Abderrahman Chafiq  
snow in the Atlas mountains from Morocco 🇲🇦🇲🇦🇲🇦🇲🇦  
Like · Reply · 1w

Dian Clayton  
Loved it, but snow on Mt Osser in November! 😊  
Like · Reply · 1w

Hany Sayed  
Beautiful  
Like · Reply · 1w

Kelly McCarthy  
Cradle Mountain is one of my favorite places. So many wild wombats to observe!!  
Like · Reply · 1w

# Ανάκτηση Πληροφορίας vs Βάσεις Δεδομένων

Δομημένα δεδομένα

Ακολουθούν κάποιο σχήμα και είναι αποθηκευμένα με βάση κάποιο μοντέλο

*Ερώτημα SQL*

```
SELECT όνομα  
FROM πλανήτες  
WHERE δορυφόροι = 0  
      OR δορυφόροι = 1  
      OR δορυφόροι = 2
```

κωδικός	όνομα	διάμετρος	δορυφόροι
1	Ερμής	4880	0
2	Αφροδίτη	12103.6	0
3	Γη	12756.3	1
4	Άρης	6794	2
5	Δίας	142984	63
6	Κρόνος	120536	34
7	Ουρανός	51118	21
8	Ποσειδώνας	49532	13
9	Πλούτωνας	2274	3

Το ερώτημα είναι σαφές, προσδιορίζει επακριβώς τη συνθήκη που πρέπει να ικανοποιεί κάθε αποτέλεσμα που εμφανίζεται στην έξοδο.

# Ανάκτηση Πληροφορίας vs Βάσεις Δεδομένων

## Συλλογή εγγράφων

d1 : Ο κομήτης του Χάλεϋ μας επισκέπτεται περίπου κάθε εβδομήντα έξι χρόνια.  
d2 : Ο κομήτης του Χάλεϋ πήρε το όνομά του από τον αστρονόμο Έντμοντ Χάλεϋ.  
d3 : Ένας κομήτης διαγράφει ελλειπτική τροχιά.  
d4 : Ο πλανήτης Άρης έχει δύο φυσικούς δορυφόρους, το Δείμο και το Φόβο.  
d5 : Ο πλανήτης Δίας έχει 63 γνωστούς φυσικούς δορυφόρους.  
d6 : Ένας κομήτης έχει μικρότερο διάμετρο από ότι ένας πλανήτης.  
d7 : Ο Άρης είναι ένας πλανήτης του ηλιακού μας συστήματος.  
...

*Πληροφοριακή ανάγκη:* πληροφορίες για τον κομήτη του Χάλεϋ

*Ερώτημα:* Χάλεϋ

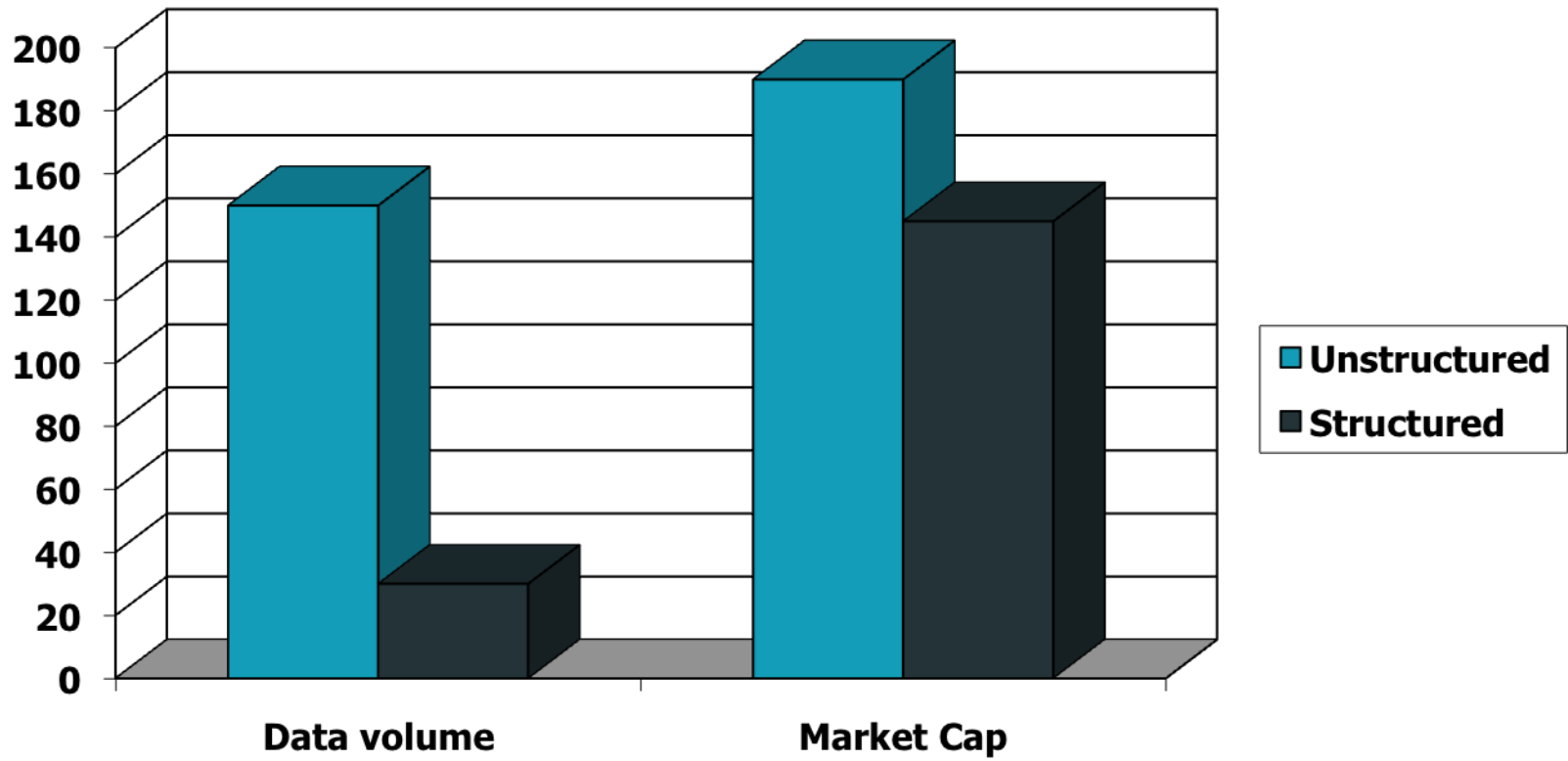
Διαισθητικά αντιλαμβανόμαστε ότι τα έγγραφα *d1* και *d2* σχετίζονται περισσότερο με το ερώτημα από ότι τα υπόλοιπα έγγραφα.

# Ημιδομημένα δεδομένα

- Στην πραγματικότητα, δεν υπάρχουν αμιγώς μη δομημένα δεδομένα
  - π.χ., αυτή η διαφάνεια έχει διακριτές ζώνες όπως *Title* και *Bullets*
  - *Web pages?*
  - *Emails?*
- «Ημιδομημένη» αναζήτηση όπως:
  - *Title* contains ημιδομημένα AND *Bullets* contain αναζήτηση

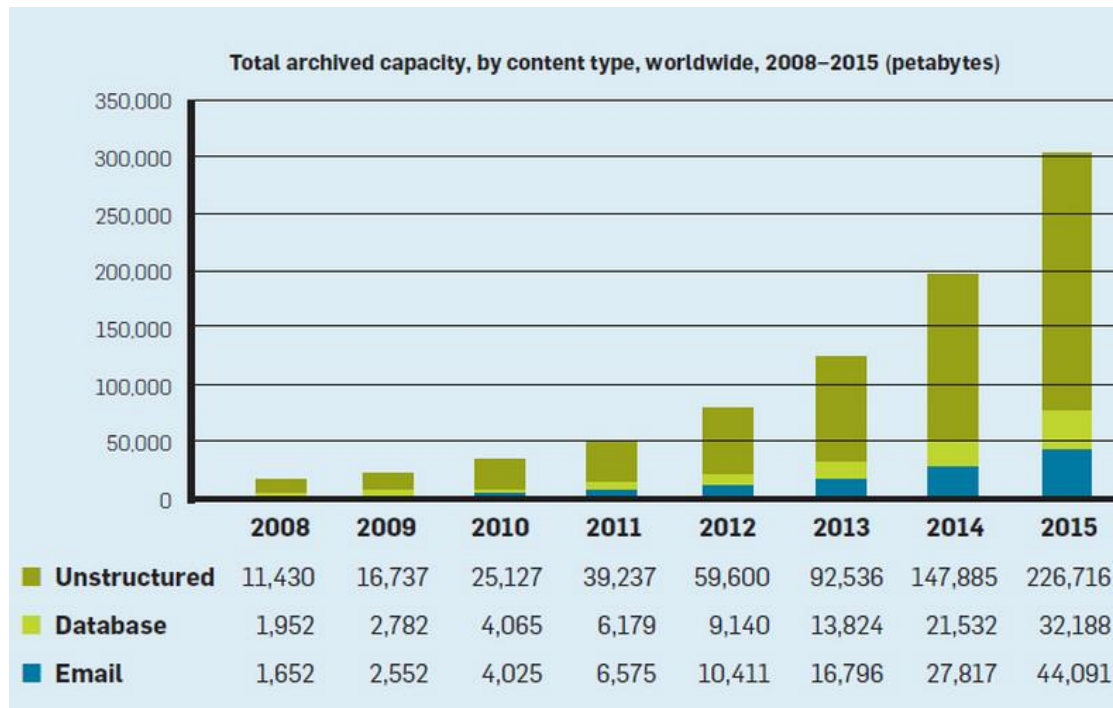
... και βέβαια υπάρχει πάντα η γλωσσική δομή

# Αδόμητα (κείμενο) vs. Δομημένα (βάσεις δεδομένων) δεδομένα το 1996





# Αδόμητα (κείμενο) vs. Δομημένα (βάσεις δεδομένων) δεδομένα σήμερα?



User generated content (social networks, blogs, etc) Example: *Facebook search*

- Ο “κόσμος” παράγει περισσότερο από **2 exabytes** νέας πληροφορίας το χρόνο, 90% της οποίας είναι σε ψηφιακή μορφή και με 50% ετήσια αύξηση

# Αναζήτηση

- Μοντελοποίηση
- Επεξεργασία (φυσικές γλώσσες)
- Βασικές δομές
- Αξιολόγηση

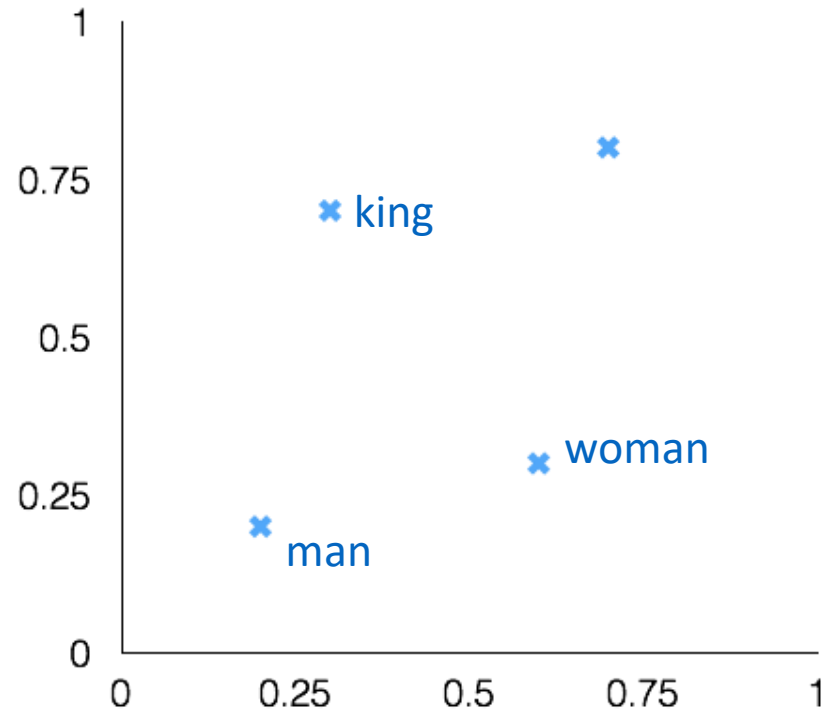
# Μηχανές αναζήτησης

- Πέρα από τους όρους
  - Συνδέσεις, Ανάλυση συνδέσμων, clickstreams ...
- Διαφορετικοί χρήστες, ανάγκες, ερωτήματα, κείμενα
- Πως δουλεύουν οι μηχανές αναζήτησης; Μπορούμε να τις βελτιώσουμε;

# Μερικά Στοιχεία Μηχανικής Μάθησης

## Word embeddings

+	king	[ 0.30 0.70 ]
-	man	[ 0.20 0.20 ]
+	woman	[ 0.60 0.30 ]
<hr/>		
	queen	[ 0.70 0.80 ]



# Άλλα θέματα

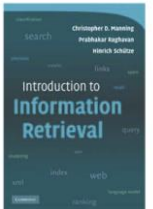
- Κατηγοριοποίηση (classification)  
Τοποθέτηση εγγράφων στη σωστή κατηγορία (Παράδειγμα: email spam)
- Συσταδοποίηση (clustering)  
Ομαδοποίηση σχετικών εγγράφων και περίληψη
- Query-Answering systems
- Συστάσεις (recommendations)
- Κριτικές (reviews)

# Διαδικαστικά

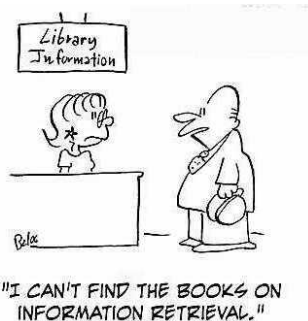
- Ιστοσελίδα

- Βιβλίο

- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze. *Εισαγωγή στην Ανάκτηση Πληροφοριών*, Εκδόσεις Κλειδάριθμος
- Η αγγλική έκδοση διαθέσιμη δωρεάν



- Ricardo Baeza-Yates and Berthier Ribeiro-Neto, *Ανάκτηση Πληροφορίας*, 2<sup>η</sup> Έκδοση, Εκδόσεις Τζιόλα



# Διαδικαστικά

- Βαθμολογία (μπορεί να αλλάξει):
  - Εργασία (έως 2 άτομα) – σε φάσεις: 50%
    - Προαιρετική εργασία – μελέτη ML μοντέλων (20%)
  - Τελικό Διαγώνισμα: 50% (αν όχι την προαιρετική εργασία)  
30% (αν την προαιρετική εργασία)
- Η εργασία δεν «κρατιέται»
- Για να περάσετε το μάθημα, βαθμός διαγωνίσματος  $\geq 4$



# Λίγα λόγια για την εργασία

Μηχανή αναζήτησης

Lucene (solar)

2 καταληκτικές ημερομηνίες

Θεματικό (πέρσι covid-related)

# Τι άλλο θα δούμε σήμερα;

1. Μερικές βασικές έννοιες

2. Ένα απλό ΣΑΠ

Μια μικρή εισαγωγή στο απλούστερο μοντέλο αναζήτησης (Boolean) (Κεφάλαιο 1 του βιβλίου)

*Ένα απλό σύστημα ΑΠ (βασικές δομές δεδομένων και παραδείγματα ερωτημάτων)*

# Boolean Ανάκτηση

# Τι είναι η Ανάκτηση Πληροφορίας (Information Retrieval);

Ανάγκη  
πληροφόρησης



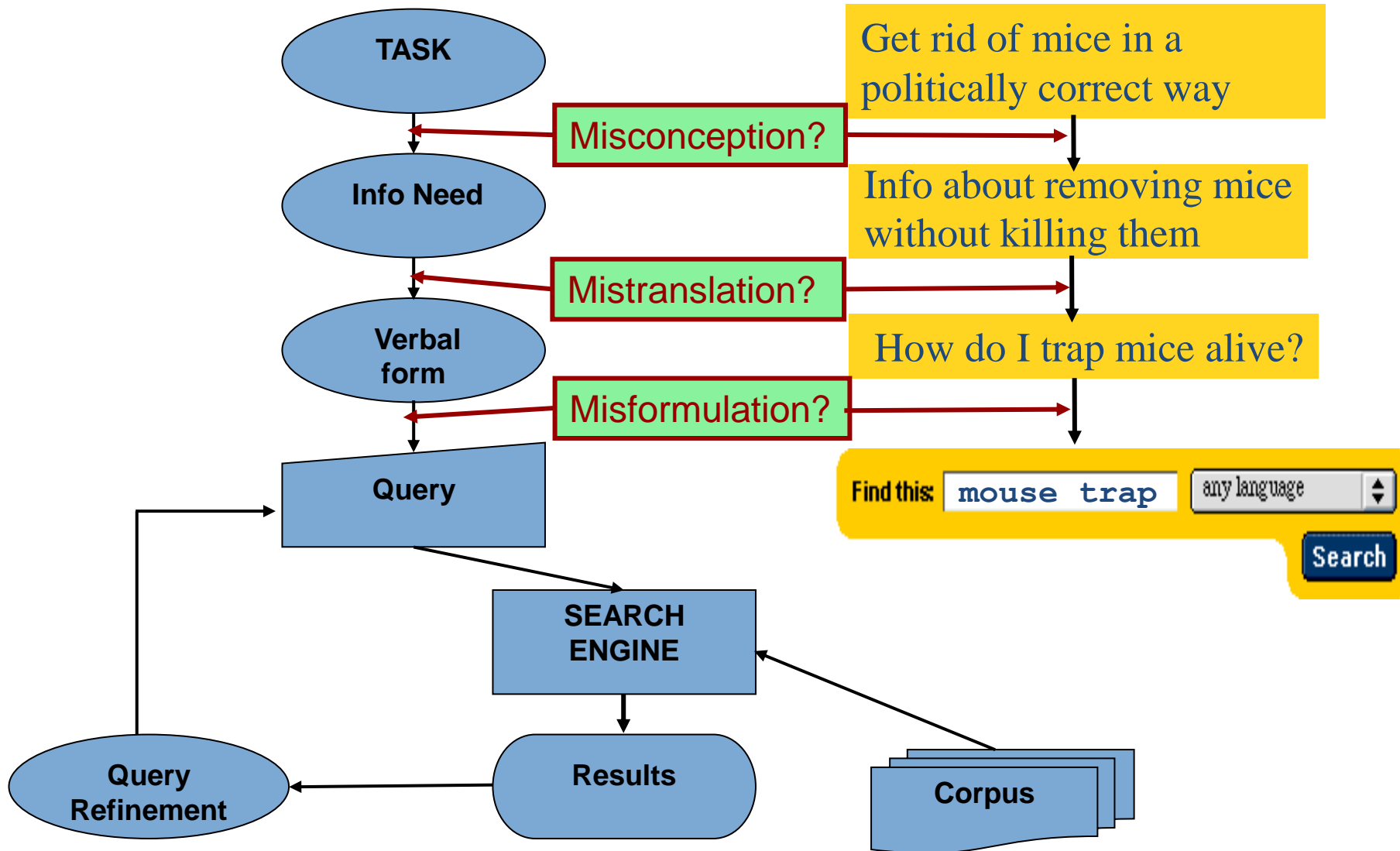
# Βασικές Έννοιες

**Συλλογή (Collection - corpus):** Σύνολο από έγγραφα

**Στόχος:** Ανάκτηση των εγγράφων που περιέχουν πληροφορία που είναι συναφής (relevant) με την ανάγκη πληροφόρησης (information need) του χρήστη και τον βοηθά να ολοκληρώσει κάποιο έργο (task)

- ✓ Διαφορά μεταξύ: information need και ερωτήματος (query)
- ✓ Ad hoc retrieval

# Το κλασικό μοντέλο αναζήτησης (search model)



# Βασικές έννοιες

Αποτέλεσμα **σε διάταξη** με βάση τη συνάφεια

**Αξιολόγηση:**

- πέρα από την απόδοση (efficiency)  
αποτελεσματικότητα (effectiveness)

**Αποτελεσματικότητα (effectiveness):** Πόσο *καλά (χρήσιμα)* είναι τα έγγραφα που ανακτήθηκαν;

# Αποτελεσματικότητα

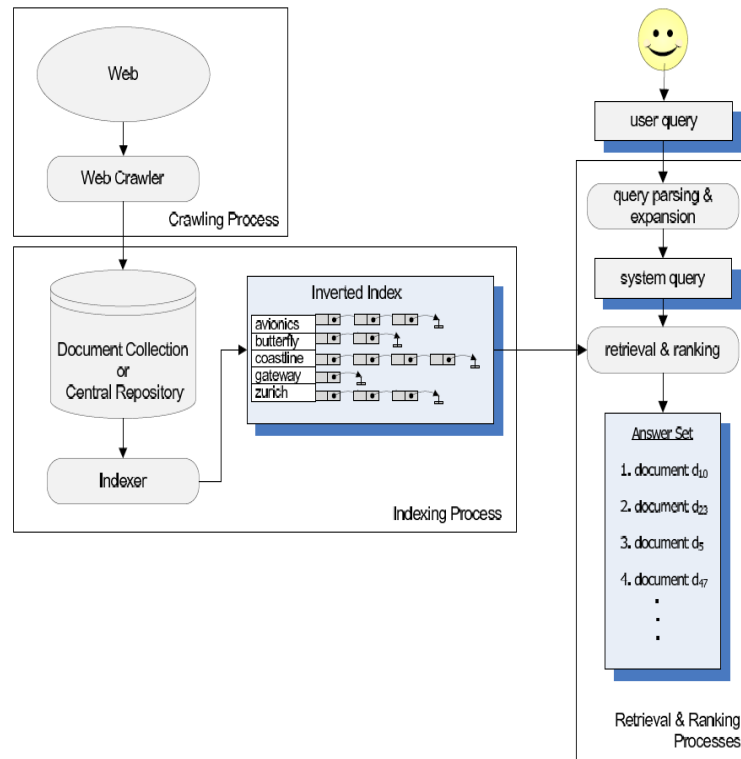
- **Ακρίβεια (Precision)**: Το ποσοστό των εγγράφων που ανακτήθηκαν που είναι συναφή με την ανάγκη πληροφόρησης του χρήστη
- **Ανάκληση (Recall)**: Το ποσοστό των συναφών με την ανάγκη πληροφόρησης του χρήστη εγγράφων της συλλογής που ανακτήθηκαν από το σύστημα
  - Περισσότερα στο μέλλον

παράδειγμα συλλογή 1000 έγγραφα  
 υπάρχουν 20 συναφή έγγραφα με το ερώτημα  
 το ΣΑΠ μας επιστρέφει 30 έγγραφα ( 17 συναφή  
 13 ης συναφής



# Architecture of the IR System

## ■ High level software architecture of an IR system



# Βασικά Βήματα

## (προεπεξεργασία)

- Σύλλεξε τα έγγραφα
- Ανάλυση
- Κατασκεύασε βοηθητικές δομές – ευρετήρια

## (λειτουργία)

- Επεξεργασία ερωτήσεων

Αρχικά θα δούμε την απλούστερη μορφή:

**Boolean retrieval**

# Για να δούμε τα βασικά ...

## Boolean μοντέλο

- Οι χρήστες διατυπώνουν ερωτήματα με τη μορφή *Boolean εκφράσεων*, δηλαδή όρων συνδυασμένων με **AND**, **OR** και **NOT**
- Επιστρέφονται ως απάντηση όλα τα κείμενα που ικανοποιούν το ερώτημα *χωρίς διάταξη*
  - *Δυαδική συνάφεια (συναφές, μη συναφές)*
- Κείμενο ως **σύνολο όρων**

# Αδόμητα δεδομένα το 1680



Shakespeare's Collected Works

# Αδόμητα δεδομένα το 1680

- Ποια θεατρικά έργα του Shakespeare περιέχουν τις λέξεις **Brutus** και **Caesar** αλλά όχι τη λέξη **Calpurnia**
  - Ερώτημα: **Brutus AND Caesar AND NOT Calpurnia**
- Να διαβάσουμε όλα τα έργα σειριακά από την αρχή σημειώνοντας ...
- Θα μπορούσαμε να κάνουμε grep σε όλα τα έργα για **Brutus** και **Caesar**, και να σβήσουμε τις γραμμές που περιέχουν τη λέξη **Calpurnia**

# Αδόμητα δεδομένα το 1680

- Γιατί όχι grep?
  - Αργό (για μεγάλες συλλογές)
  - Grep line-oriented, η ανάκτηση πληροφορίας **document-oriented**
  - ***NOT Calpurnia*** δεν είναι εύκολο
  - Επιπρόσθετη λειτουργικότητα (π.χ., βρες τη λέξη ***Romans*** κοντά στο ***countrymen***)
  - Διάταξη! Ranked retrieval (τα «καλύτερα» έγγραφα ανάμεσα σε αυτά που ικανοποιούν την ερώτηση)
    - Σε επόμενα μαθήματα ....

Θα προ-επεξεργαστούμε τα έγγραφα και θα δημιουργήσουμε ευρετήρια

Για να δούμε τα βασικά ...

## Boolean μοντέλο

Δυαδική μήτρα (πίνακας) σύμπτωσης  $M$

Γραμμές: **Term** (όροι, λέξεις)

Στήλες: **Document** (έγγραφα, έργα)

$M[i, j] = 1$ , αν ο όρος  $i$  εμφανίζεται στο έγγραφο  $j$   
0, αλλιώς

# Term-document incidence matrix (μήτρα σύμπτωσης)

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

***Brutus AND Caesar BUT NOT Calpurnia***

1 αν το **έργο** περιέχει τη **λέξη**, 0 αλλιώς



# Οι όροι και τα έγγραφα ως διανύσματα

Έχουμε ένα *δυαδικό διάνυσμα* για κάθε *όρο* και κάθε *έγγραφο*

- Για να απαντήσουμε στην ερώτηση: παίρνουμε τα διανύσματα για το ***Brutus, Caesar*** και το συμπλήρωμα του διανύσματος για το ***Calpurnia*** → bitwise *AND*.

$$110100 \text{ AND } 110111 \text{ AND } 101111 = 100100.$$

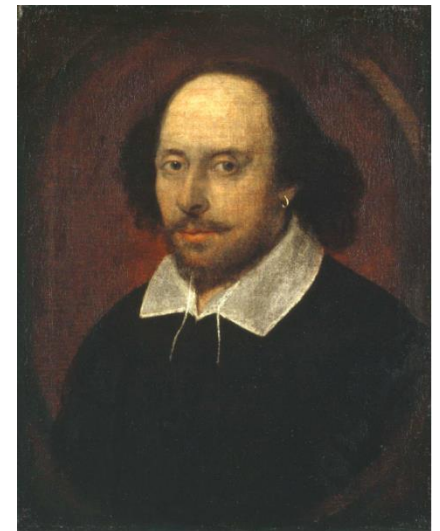
## Οι απαντήσεις:

### • Antony and Cleopatra, Act III, Scene ii

*Agrippa* [Aside to DOMITIUS ENOBARBUS]: Why, Enobarbus,  
When Antony found Julius **Caesar** dead,  
He cried almost to roaring; and he wept  
When at Philippi he found **Brutus** slain.

### • Hamlet, Act III, Scene ii

*Lord Polonius*: I did enact Julius **Caesar** I was killed i' the  
Capitol; **Brutus** killed me.



# παράδειγμα

$d_1$  a b c a

$d_2$  d a b

$d_3$  b a f

$d_4$  a e

$d_5$  d f

# Μεγαλύτερες συλλογές

- Ας θεωρήσουμε  $N = 1$  εκατομμύρια έγγραφα, το καθένα έχει περίπου 1000 (διακριτούς) όρους (~2-3 σελίδες βιβλίου).
- Έστω ότι ανάμεσα τους υπάρχουν  $M = 500K$  διακριτοί (*distinct*) όροι.

Πόσα κελιά έχει ο πίνακας σύμπτωσης;

- A. 1 δισεκατομμύριο
- B. 500 δισεκατομμύρια
- C. 500 εκατομμύρια
- D. 5 δισεκατομμύρια

Πόσα μη μηδενικά κελιά (δηλαδή 1) έχει ο πίνακας σύμπτωσης;

- A. 1 δισεκατομμύριο
- B. 1 εκατομμύριο
- C. 50 δισεκατομμύρια
- D. 50 εκατομμύρια

Ποσοστό των 1:

## Πόσο είναι το μέγεθος του πίνακα;

- Ο 500K x 1M πίνακας έχει μισό τρισεκατομμύριο 0's και 1.
- Αλλά δεν έχει περισσότερα από ένα δισεκατομμύριο 1.
  - Ο πίνακας είναι εξαιρετικά **αραιός** (sparse) – τουλάχιστον το 99.8% είναι 0.
- Ποια είναι μια καλύτερη αναπαράσταση;

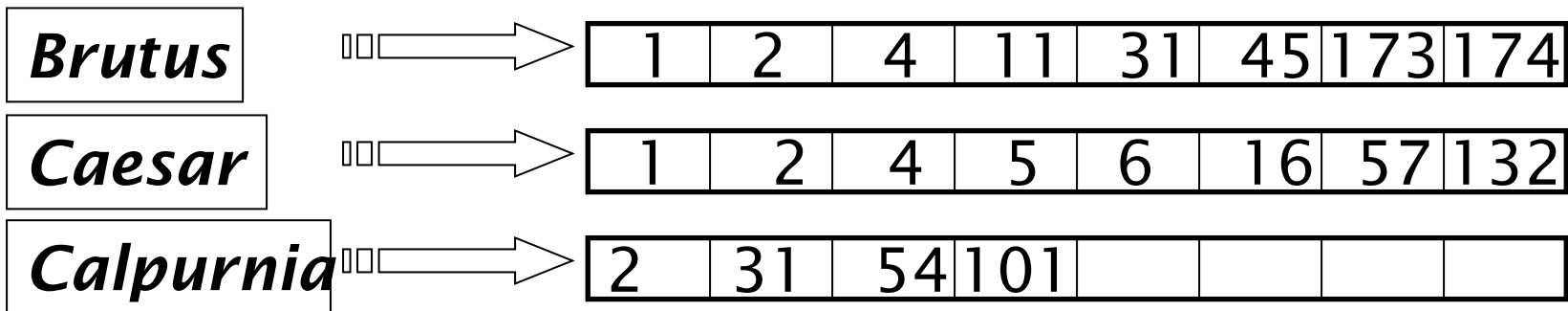
# Αντεστραμμένο ευρετήριο ή αρχείο (Inverted index/file)

Για κάθε *όρο (term) t*, διατηρούμε μια λίστα με όλα τα έγγραφα που περιέχουν τον όρο.

- Κάθε έγγραφο χαρακτηρίζεται από ένα **αναγνωριστικό εγγράφου (docID)**, πχ αριθμό που ανατίθεται σειριακά στα έγγραφα κατά τη δημιουργία τους

## Αντεστραμμένο ευρετήριο

- Μπορούμε να χρησιμοποιήσουμε σταθερού μεγέθους arrays για αυτό?



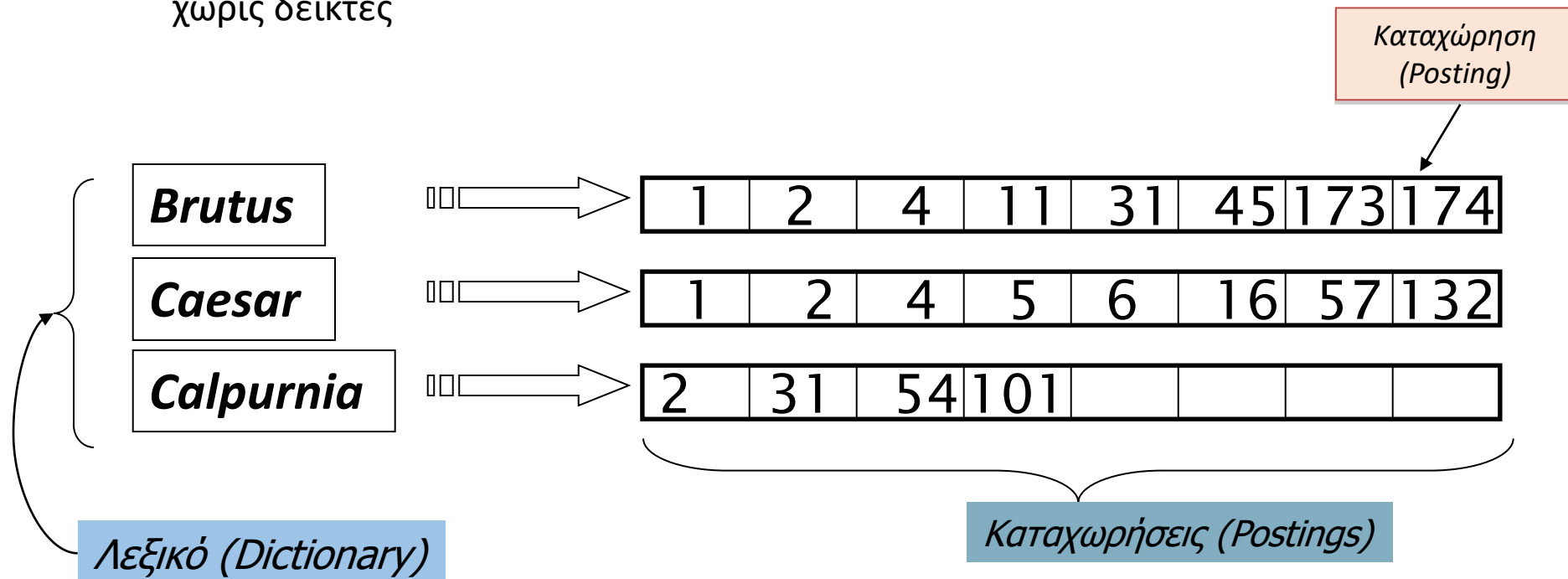
Τι γίνεται αν η λέξη **Caesar** προστεθεί στο έγγραφο 14?

# Αντεστραμμένο ευρετήριο

- Χρειαζόμαστε μεταβλητού μεγέθους **λίστες καταχωρήσεων (postings lists)**

Ποια δομή δεδομένων είναι κατάλληλη;

- Στη μνήμη, απλά-διασυνδεδεμένες λίστες (skip lists) ή πίνακες μεταβλητού μήκους
- Στο δίσκο, ως (συμπιεσμένες) συνεχόμενες ακολουθίες καταχωρήσεων χωρίς δείκτες



Σε διάταξη με βάση το docID (θα δούμε σε λίγο γιατί!).



# παράδειγμα

$d_1$  a b c a

$d_2$  d a b

$d_3$  b a f

$d_4$  a e

$d_5$  d f

# Βασική Ορολογία

- **Αντεστραμμένο ευρετήριο** (Inverted index)
- **Λίστες καταχωρήσεων** (posting lists) – μία για κάθε όρο
  - Καταχώρηση – ένα στοιχείο της λίστας
  - ✓ Κάθε λίστα είναι διατεταγμένη με το DocID
- **Λεξιλόγιο** (Vocabulary): το σύνολο των όρων
- **Λεξικό** (Dictionary) δομή δεδομένων για τους όρους
  - ✓ Αρχικά ας θεωρήσουμε αλφαβητική διάταξη

*Το δημιουργούμε από πριν, θα δούμε πως*

# Κατασκευή του αντεστραμμένου ευρετηρίου

Έγγραφα προς  
ευρετηριοποίηση



Friends, Romans, countrymen.  
⋮

Tokenizer

Token stream

Friends

Romans

Countrymen

Θα τα δούμε σε  
επόμενα  
μαθήματα.

Linguistic  
modules

Γλωσσολογική επεξεργασία

friend

roman

countryman

Modified tokens

Indexer

*friend*



2

4

*roman*



1

2

*countryman*



13

16

Inverted index

# Βήματα του Indexer: Ακολουθία Token

- Ακολουθία από ζεύγη (Modified token, Document ID).

Doc 1

I did enact Julius  
Caesar I was killed  
i' the Capitol;  
Brutus killed me.

Doc 2

So let it be with  
Caesar. The noble  
Brutus hath told you  
Caesar was ambitious



Term	docID
I	1
did	1
enact	1
julius	1
caesar	1
I	1
was	1
killed	1
i'	1
the	1
capitol	1
brutus	1
killed	1
me	1
so	2
let	2
it	2
be	2
with	2
caesar	2
the	2
noble	2
brutus	2
hath	2
told	2
you	2
caesar	2
was	2
ambitious	2

# Βήματα του Indexer: Ταξινόμηση (sort)

- **Ταξινόμηση** με βάση τους όρους

- Και μετά το docID



Βασικό βήμα της  
ευρετηριοποίησης

Term	docID
I	1
did	1
enact	1
julius	1
caesar	1
I	1
was	1
killed	1
i'	1
the	1
capitol	1
brutus	1
killed	1
me	1
so	2
let	2
it	2
be	2
with	2
caesar	2
the	2
noble	2
brutus	2
hath	2
told	2
you	2
caesar	2
was	2
ambitious	2



Term	docID
ambitious	2
be	2
brutus	1
brutus	2
capitol	1
caesar	1
caesar	2
caesar	2
did	1
enact	1
hath	1
I	1
I	1
i'	1
it	2
julius	1
killed	1
killed	1
let	2
me	1
noble	2
so	2
the	1
the	2
told	2
you	2
was	1
was	2
with	2

# Βήματα του Indexer: Λεξικό & Καταχωρήσεις

- Πολλαπλές εμφανίσεις του όρου σε ένα έγγραφο συγχωνεύονται (merged).
- Διαχωρισμός σε *λεξικό* και *καταχωρήσεις*
- Προσθέτουμε και πληροφορία για τη συχνότητα εγγράφου (doc. frequency).

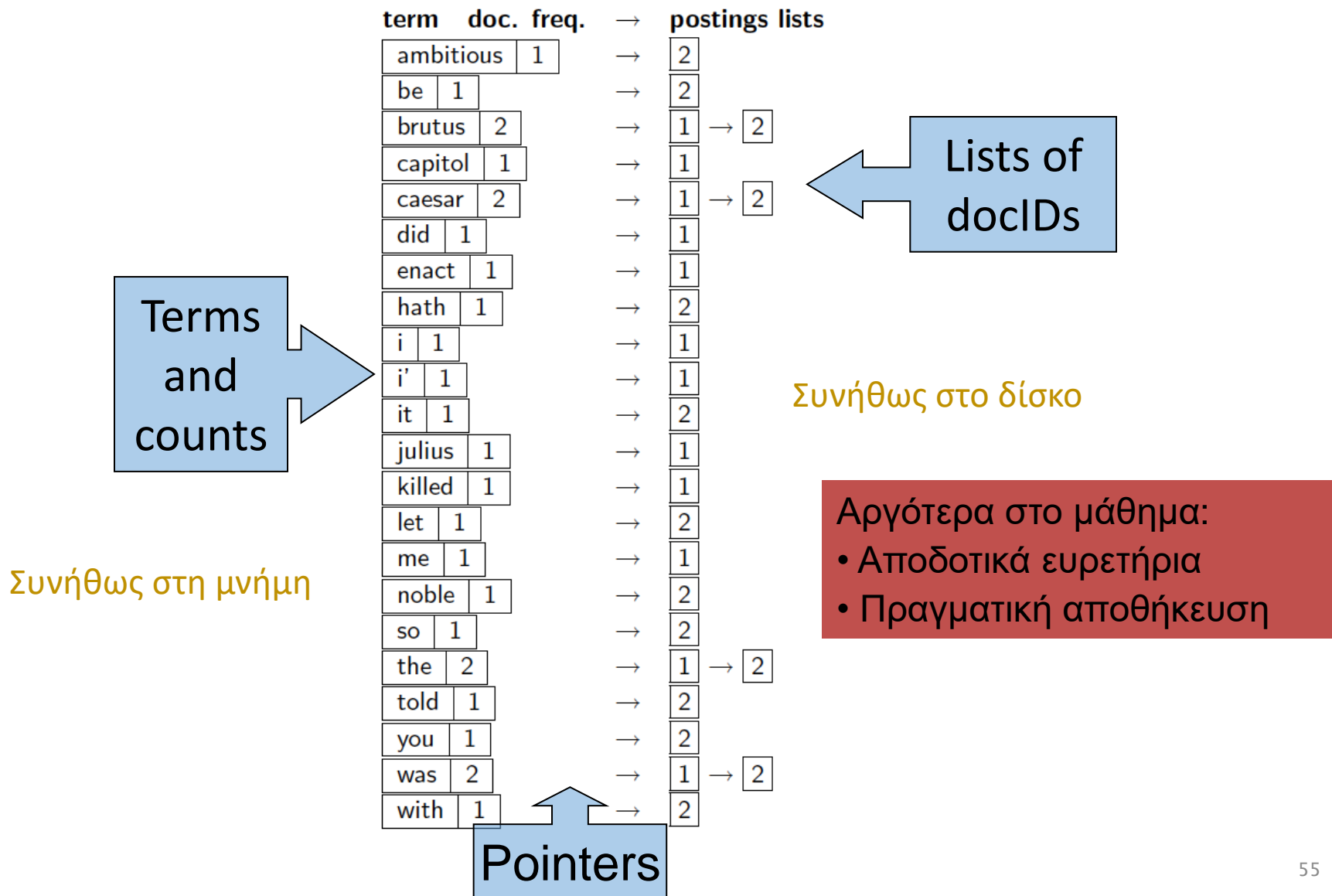
Γιατί τη συχνότητα;  
Επίσης, συχνότητα όρου (term frequency)

Term	docID
ambitious	2
be	2
brutus	1
brutus	2
capitol	1
caesar	1
caesar	2
caesar	2
did	1
enact	1
hath	1
I	1
I	1
i'	1
it	2
julius	1
killed	1
killed	1
let	2
me	1
noble	2
so	2
the	1
the	2
told	2
you	2
was	1
was	2
with	2



term	doc. freq.	→	postings lists
ambitious	1	→	[2]
be	1	→	[2]
brutus	2	→	[1] → [2]
capitol	1	→	[1]
caesar	2	→	[1] → [2]
did	1	→	[1]
enact	1	→	[1]
hath	1	→	[2]
i	1	→	[1]
i'	1	→	[1]
it	1	→	[2]
julius	1	→	[1]
killed	1	→	[1]
let	1	→	[2]
me	1	→	[1]
noble	1	→	[2]
so	1	→	[2]
the	2	→	[1] → [2]
told	1	→	[2]
you	1	→	[2]
was	2	→	[1] → [2]
with	1	→	[2]

# Πόσο χώρο χρειαζόμαστε?



## Φτιάξαμε το ευρετήριο, τώρα;

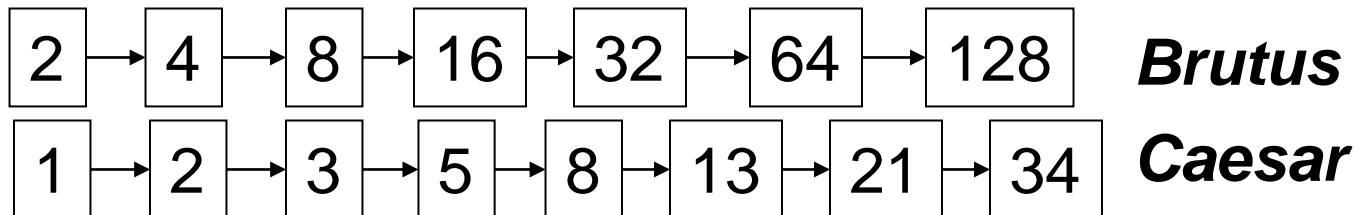
- Πως επεξεργαζόμαστε μια ερώτηση;
  - Αργότερα – τι άλλου είδους ερωτήσεις



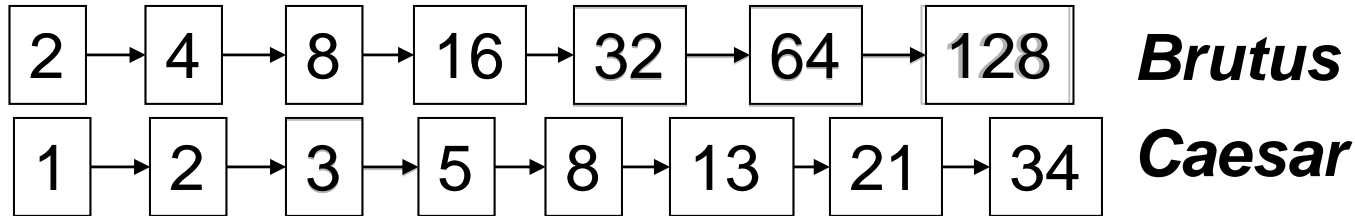
# Επεξεργασία ερωτήσεων: AND

Έστω η ερώτηση: **Brutus AND Caesar**

- Βρες το **Brutus** στο Λεξικό
  - Ανέκτησε τις καταχωρήσεις.
- Βρες το **Caesar** στο Λεξικό
  - Ανέκτησε τις καταχωρήσεις.
- Υπολογισμό της τομής του – ΠΩΣ;



- Διέσχισε τις δύο λίστες ταυτόχρονα, σε χρόνο γραμμικό (linear) στο συνολικό αριθμό των καταχωρήσεων



Αν τα μήκη των λιστών είναι  $x$  και  $y$ , η συγχώνευση παίρνει  $O(x+y)$  λειτουργίες.  
Σημαντικό: οι καταχωρήσεις πρέπει να είναι διατεταγμένες με βάση το docID.

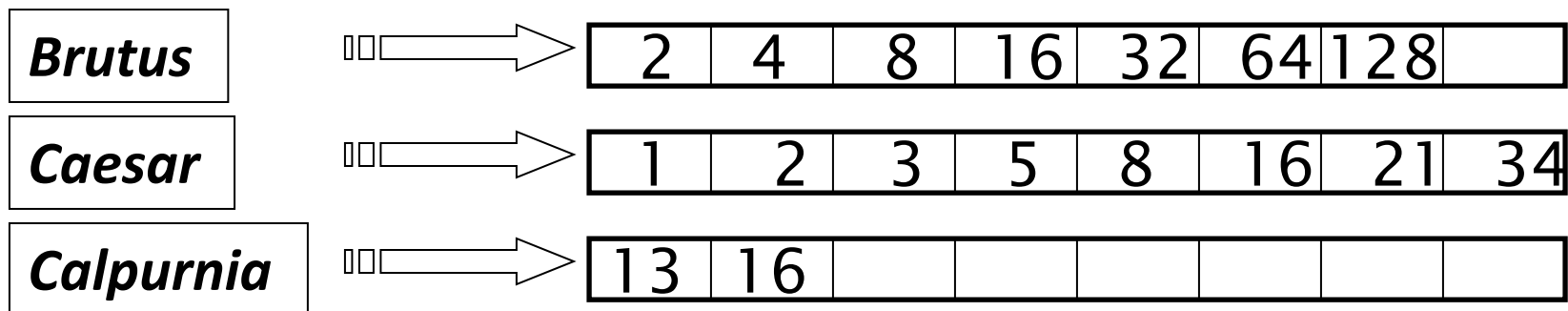
## Ο αλγόριθμος συγχώνευσης

INTERSECT( $p_1, p_2$ )

```
1  answer  $\leftarrow \langle \rangle$ 
2  while  $p_1 \neq \text{NIL}$  and  $p_2 \neq \text{NIL}$ 
3  do if  $\text{docID}(p_1) = \text{docID}(p_2)$ 
4      then  $\text{ADD}(\textit{answer}, \text{docID}(p_1))$ 
5           $p_1 \leftarrow \text{next}(p_1)$ 
6           $p_2 \leftarrow \text{next}(p_2)$ 
7      else if  $\text{docID}(p_1) < \text{docID}(p_2)$ 
8          then  $p_1 \leftarrow \text{next}(p_1)$ 
9          else  $p_2 \leftarrow \text{next}(p_2)$ 
10 return answer
```

# Βελτιστοποίηση ερωτήματος

- Ποια είναι βέλτιστη σειρά για την επεξεργασία ενός ερωτήματος;
- Έστω μια ερώτηση που είναι το *AND*  $n$  όρων.
- Για καθέναν από τους  $n$  όρους, βρες τις καταχωρήσεις του και εκτέλεσε το *AND* σε όλες.

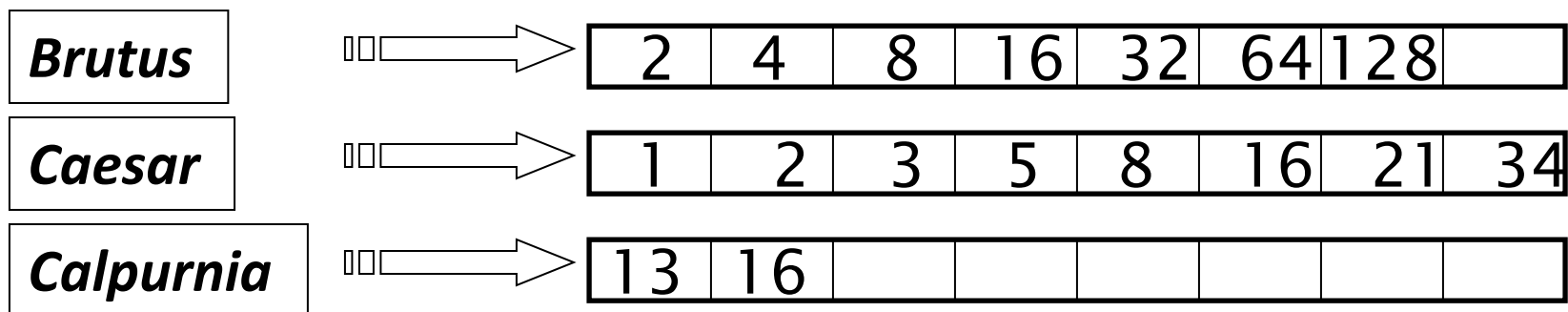


**Query: Brutus AND Calpurnia AND Caesar**

# Βελτιστοποίηση ερωτήματος

- Επεξεργασία με αύξουσα συχνότητα:
  - Ξεκίνησε με το *μικρότερο* σύνολο και συνέχισε μειώνοντας και άλλο το αποτέλεσμα

Χρήση της συχνότητας εγγράφου  
στο λεξικό



Εκτέλεση του ερωτήματος ως ***(Calpurnia AND Brutus) AND Caesar***.

## Βελτιστοποίηση ερωτήματος

Π.χ., (*madding OR crowd*) AND (*ignoble OR strife*)

- Βρες τη συχνότητα εγγράφου για όλους τους όρους.
- Εκτίμησε το μέγεθος κάθε *OR* (συντηρητικά: ως το άθροισμα των συχνοτήτων εγγράφου).
- Επεξεργασία του ερωτήματος κατά αύξουσα σειρά κάθε όρου.

# Βελτιστοποίηση ερωτήματος

((A and B) and C) and D

- Κρατάμε *το ενδιάμεσο αποτέλεσμα στη μνήμη* και διαβάζουμε τη άλλη λίστα από το δίσκο
- Αρχικά, ενδιάμεσο αποτέλεσμα = A

Όταν πολλοί μεγάλες λίστες, εναλλακτικές για τον υπολογισμό τομής

- χρησιμοποιώντας δυαδική αναζήτηση στη μεγάλη λίστα (λογαριθμικός χρόνος)
- αποθήκευση μεγάλης λίστας ως hashtable (σταθερά)

❖ Δοκιμάστε το <http://www.rhymezone.com/shakespeare/>



## Boolean ερωτήματα: Ακριβές ταίριασμα (Exact match)

- Το **Boolean μοντέλο ανάκτησης** απαντά ερωτήματα που είναι Boolean εκφράσεις:
  - Χρήση *AND*, *OR* και *NOT* για το συνδυασμό όρων
    - Θεωρούν κάθε έγγραφο ως ένα **σύνολο** όρων
    - Η αναζήτηση είναι ακριβής (precise): *ένα έγγραφο είτε ικανοποιεί τη συνθήκη είτε όχι.*
  - Ίσως, το απλούστερο μοντέλο
- Το βασικό μοντέλο σε εμπορικά συστήματα για 3 δεκαετίες (πριν τον web).
- Πολλά συστήματα ακόμα Boolean:
  - Email, library catalog, Mac OS X Spotlight

Η Google χρησιμοποιεί το Boolean μοντέλο ?

Άλλα ερωτήματα

## Παράδειγμα: WestLaw

<https://legal.thomsonreuters.com/en/products/westlaw>

- Μεγάλο εμπορικό (συνδρομές επί πληρωμή) σύστημα
- Αναζήτηση σε νομικά κείμενα (άρχισε το 1975, η διάταξη προστέθηκε το 1992)
- Boolean ερωτήματα και ερωτήματα φυσικής γλώσσας
- 40,000 databases for case laws, 60 countries (source: Wikipedia)

## Παράδειγμα: WestLaw

- Παράδειγμα (απόσταση ανάμεσα στους όρους):
  - *Ανάγκη πληροφόρησης*: What is the statute of limitations in cases involving the federal tort claims act?
  - *Ερώτημα*:  
LIMIT! /3 STATUTE ACTION /S FEDERAL /2 TORT /3 CLAIM
    - /3 = within 3 words, /S = in same sentence
  
- Παράδειγμα (ανεκτική ανάκτηση):
  - *Ανάγκη πληροφόρησης*: Information on the legal theories involved in preventing the disclosure of trade secrets by employees formerly employed by a competing company
  - *Ερώτημα*:  
“trade secret” /s disclos! /s prevent /s employe!

## Παράδειγμα: WestLaw

- Ακόμα ένα παράδειγμα:
  - Requirements for disabled people to be able to access a workplace
  - *disabl! /p access! /s work-site work-place (employment /3 place)*
- SPACE σημαίνει διάζευξη (disjunction)
- *Μακροσκελή, επακριβή ερωτήματα, τελεστές εγγύτητας (proximity operators), διατυπωμένα σταδιακά (διαφορά από web search)*
- *Boolean αναζήτηση χρησιμοποιείται ακόμα από πολλούς επαγγελματίες*
  - Ξέρεις ακριβώς τι παίρνεις ως απάντηση
- Αυτό δε σημαίνει ότι δουλεύει καλύτερα

# Evidence accumulation

- 1 vs. 0 εμφάνιση ενός όρου αναζήτησης
  - 2 vs. 1 εμφανίσεις
  - 3 vs. 2 εμφανίσεις, κλπ.
  - Συχνά φαίνεται καλύτερο
- Χρειαζόμαστε και τη συχνότητα εμφάνισης του όρου στα έγγραφα

# Ποιο περίπλοκη ημιδομημένη αναζήτηση

- *Title* is about Object Oriented Programming AND Author something like stro\*rup
  - όπου \* είναι ο wild-card τελεστής
- Θέματα:
  - Πως αντιμετωπίζουμε το “about”?
  - Πως γίνεται η κατάταξη?

# Τι είδαμε σήμερα

1- Βασικές έννοιες

2- Είδαμε ένα απλό σύστημα ανάκτησης πληροφορίας  
βασισμένο στο Boolean μοντέλο

α- ανάλυση εγγράφου

β- κατασκευάζουμε το ανεστραμμένο ευρετήριο

γ- το χρησιμοποιούμε για να απαντάμε σε ερωτήματα



# ΤΕΛΟΣ 1<sup>ου</sup> Κεφαλαίου

Ερωτήσεις?

*Χρησιμοποιήθηκε κάποιο υλικό των:*

- ✓ *Pandu Nayak and Prabhakar Raghavan, CS276:Information Retrieval and Web Search (Stanford)*
- ✓ *Απόστολου Ν. Παπαδόπουλου , Ανάκτηση Πληροφορίας (Τμήμα Πληροφορικής, Αριστοτέλειο Πανεπιστήμιο)*