Information Retrieval (Unit-1)

Search Engine and Information Retrieval -

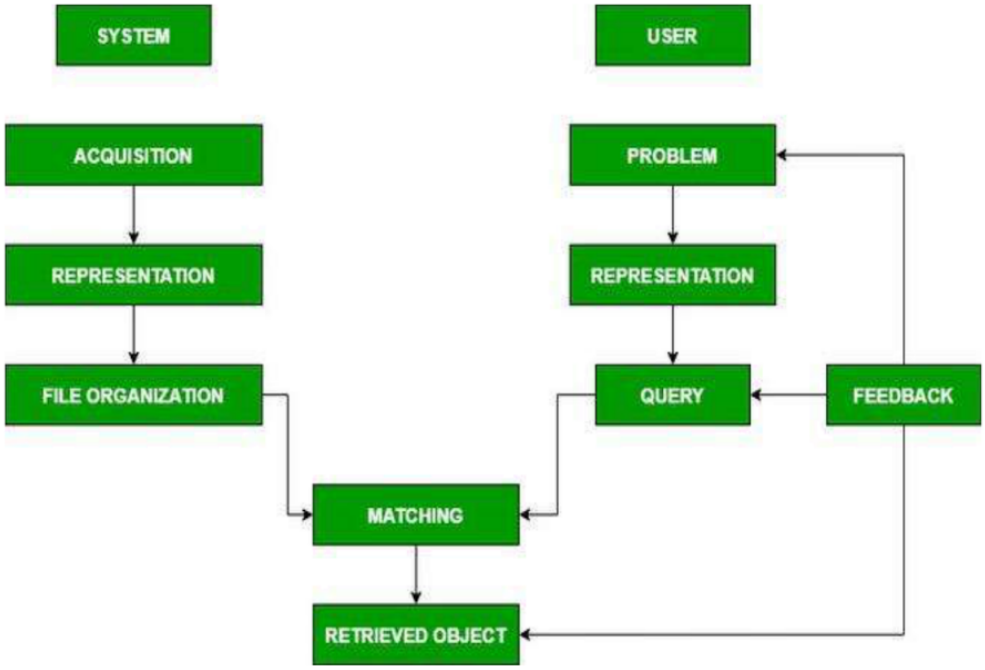
<u>Types of Search</u> :

1) Vertical Search - It is a specialized form of web search where the domain of the search is restricted to a particular topic.
2) Enterprise Search - It involves Finding the required information in huge variety of computer files scattered across a corporate internet.
3) Desktop Search - It is the personal version of the enterprise search where the information sources are the files stored on an individual computer including email messages and web pages that have recently been browsed.
4) Peer-To-Peer Search - It involves finding information in networks of nodes or computers without any centralized control.
5) Ad Hoc Search - It includes text based tasks, filtering, classification and question answering.

- Information retrieval is the activity of obtaining material which can usually be documents of an unstructured nature.
- Information Retrieval is the dominant form of information access.
- IR has the ability to represent, store, organize and access information items.

Information Retrieval	Data Retrieval
Information retrieval deals with the organization, storage, retrieval, and evaluation of information from document repositories, particularly textual information.	Data retrieval deals with obtaining data from a database management system such as ODBMS. It is A process of identifying and retrieving the data from the database, based on the query provided by the user or application.
Retrieves information about a subject.	Determines the keywords in the user query and retrieves the data.
Small errors are likely to go unnoticed.	A single error object means total failure.
Not always well structured and is semantically ambiguous.	Has a well-defined structure and semantics.
Does not provide a solution to the user of the database system.	Provides solutions to the user of the database system.
The results obtained are approximate matches.	The results obtained are exact matches.
Results are ordered by relevance.	Results are unordered by relevance.
It is a probabilistic model.	It is a deterministic model.

Components of Information Retrieval / IR Model -



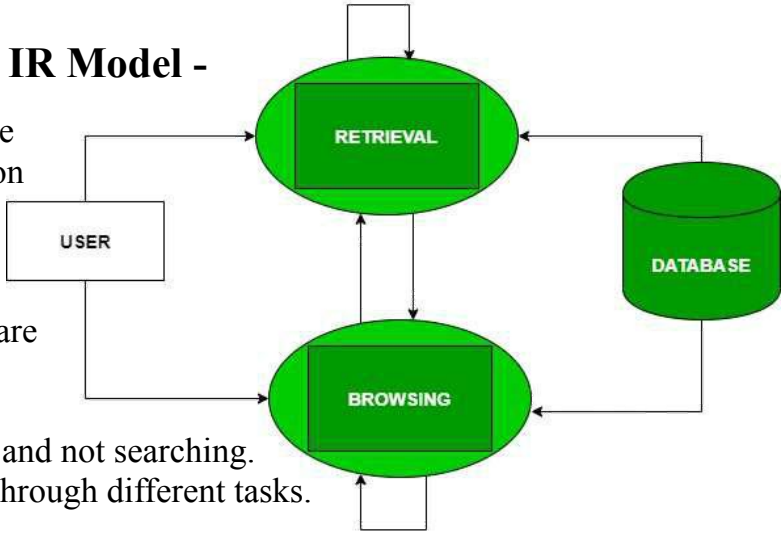
Acquisition - In this step, the selection of documents and other objects from various web resources that consist of text-based documents takes place. The required data is collected by web crawlers and stored in the database.

Representation - It consists of indexing that contains free-text terms, controlled vocabulary, manual & automatic techniques as well. example: Abstracting contains summarizing and Bibliographic description that contains author, title, sources, data, and metadata.

File Organization - There are two types of file organization methods. i.e. Sequential: It contains documents by document data. Inverted: It contains term by term, list of records under each term. Combination of both.

Query - An IR process starts when a user enters a query into the system. Queries are formal statements of information needs, for example, search strings in web search engines. In information retrieval, a query does not uniquely identify a single object in the collection. Instead, several objects may match the query, perhaps with different degrees of relevance.

Components of Information Retrieval / IR Model -



User Task - The information first is supposed to be translated into a query by the user. In the information retrieval system, there is a set of words that convey the semantics of the information that is required whereas, in a data retrieval system, a query expression is used to convey the constraints which are satisfied by the objects. Example: A user wants to search for something but ends up searching with another thing. This means that the user is browsing and not searching. The above figure shows the interaction of the user through different tasks.

Logical View of the Documents - A long time ago, documents were represented through a set of index terms or keywords. Nowadays, modern computers represent documents by a full set of words which reduces the set of representative keywords. This can be done by eliminating stop words i.e. articles and connectives. These operations are text operations. These text operations reduce the complexity of the document representation from full text to a set of index terms.

Past, Present, and Future of Information Retrieval -

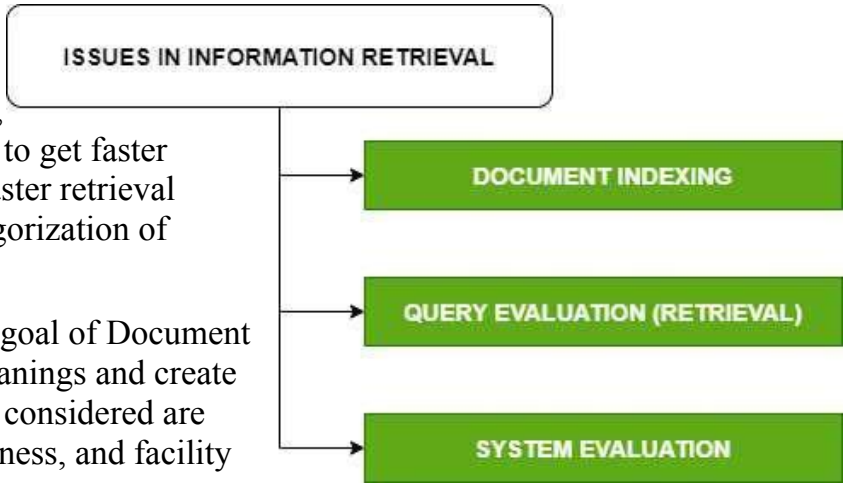
Early Developments - As there was an increase in the need for a lot of information, it became necessary to build data structures to get faster access. The index is the data structure for faster retrieval of information. Over centuries manual categorization of hierarchies was done for indexes.

Information Retrieval In Libraries - Libraries were the first to adopt IR systems for information retrieval. In first-generation, it consisted, automation of previous technologies, and the search was based on author name and title. In the second generation, it included searching by subject heading, keywords, etc. In the third generation, it consisted of graphical interfaces, electronic forms, hypertext features, etc.

The Web and Digital Libraries - It is cheaper than various sources of information, it provides greater access to networks due to digital communication and it gives free access to publish on a larger medium.

Issue Related to Information Retrieval -

The main issues of the Information Retrieval (IR) are Document and Query Indexing, Query Evaluation, and System Evaluation.



Early Developments - As there was an increase in the need for a lot of information, it became necessary to build data structures to get faster access. The index is the data structure for faster retrieval of information. Over centuries manual categorization of hierarchies were done for indexes.

Document and Query Indexing - Main goal of Document and Query Indexing is to find important meanings and create an internal representation. The factors to be considered are accuracy to represent semantics, exhaustiveness, and facility for a computer to manipulate.

Query Evaluation - In the retrieval model how can a document be represented with the selected keywords and how are documents and query representations compared to calculate a score. Information Retrieval (IR) deals with issues like uncertainty and vagueness in information systems.

Uncertainty - The available representation does not typically reflect true semantics of objects such as images, videos etc.

Vagueness - The information that the user requires lacks clarity, is only vaguely expressed in a query, feedback or user action.

System Evaluation - System Evaluation tells about the importance of determining the impact of information given on user achievement. Here, we see if the efficiency of the particular system related to time and space.

Boolean Retrieval Model -

In the Boolean Retrieval model, we give any query in the form of Boolean expression of terms which are combined with operators AND,OR and NOT.

Basic Terms -

1) Documents - It is the unit of information that we want to return as a result of a query. For example news article

2) Collection - It is the group of documents that retrieval is performed on it. for e.g Wikipedia

3) Term- It is the smallest unit of information in a query. For e.g token

4) Information Need - It is the topic about which the user desires to know more and is differentiated from a query.

5) Query - User conveys to the computer in an attempt to communicate the information needed.

6) Inverted Index - Also called an inverted file. It is an index which always maps index back from terms to the parts of documents where they occur.

7) Posting - Each item in the list which records that a term appeared in a document.

Example -

Doc 1
I did enact Julius Caesar: I was killed i' the Capitol; Brutus killed me.

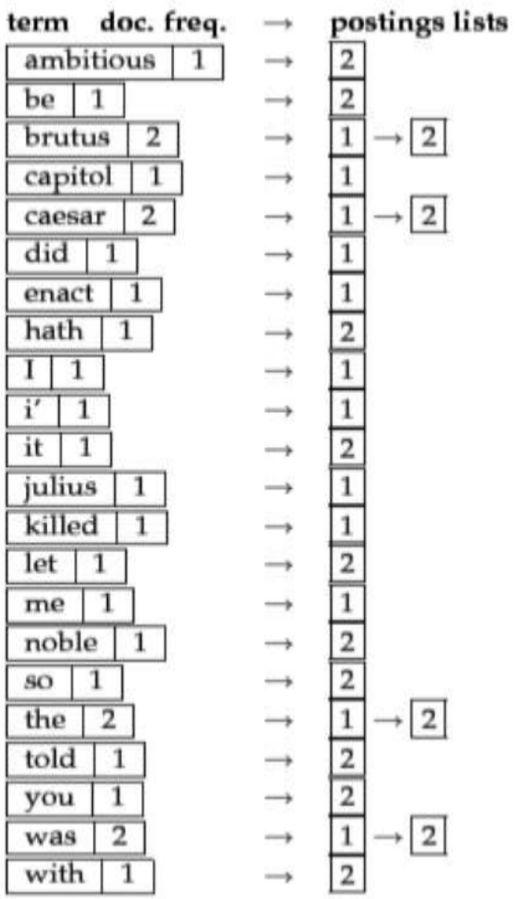
Doc 2
So let it be with Caesar. The noble Brutus hath told you Caesar was ambitious:

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

⟹

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

⟹



Extended Boolean Model VS Ranked Retrieval

- The Boolean retrieval model is opposite of ranked retrieval models like the vector space model where users use free text queries where we type one or more words rather than using precise language operators to build up query expressions.
- A proximity operator means to specify two terms in an operator query that occur close to each other in a document , where closeness may be measured by limiting the allowed number of intervening words or by reference to a structural unit such as a sequence or paragraph.

Dictionaries VS Tolerant Retrieval

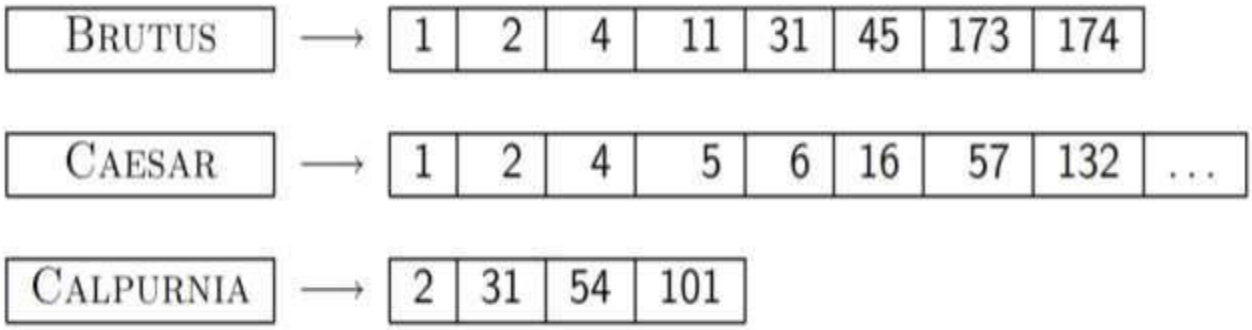
- **Dictionary Data Structure -**
  - A dictionary is a general-purpose data structure for storing a group of objects.
  - A dictionary has a set of keys and each key has a single associated value.
  - A dictionary is also called a hash, a map, a hashmap in different programming languages.
  - The dictionary data structure stores the term vocabulary, document frequency, and pointers to each postings list.

Main Operations of Dictionaries -

Dictionaries typically support so many operations -

- Retrieve a value (based on language, attempting to retrieve a missing key may provide a default value or throw an exception)
- Inserting or updating a value (typically, if the key does not exist in the dictionary, the key-value pair is inserted; if the key already exists, its corresponding value is overwritten with the new one)
- Remove or delete a key-value pair
- Test or verify for existence of a key

Example -



To implement Dictionary Data Structure -

There Two Main Choices as follows :

1) Hash Table –

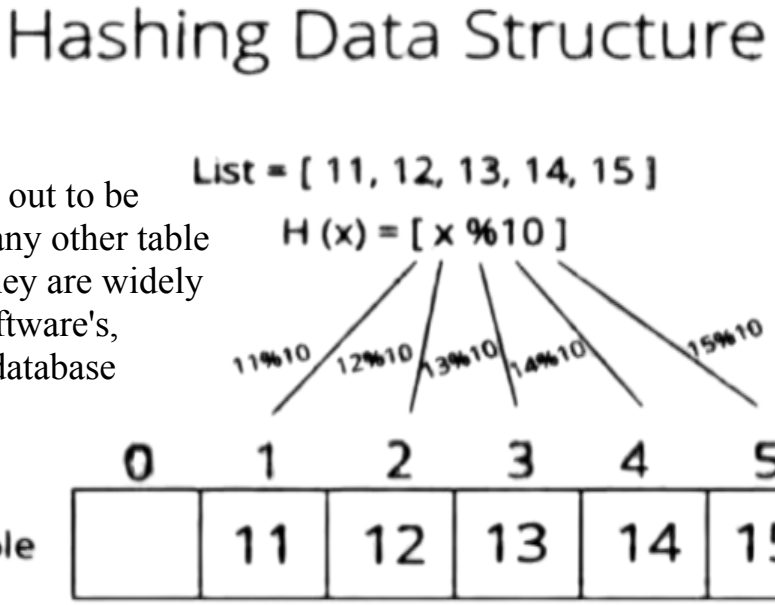
- Hash Table is a data structure which stores data in an associative manner.
- In a hash table, data is stored in an array format, where each data value has its own unique index value.
- Access of data becomes very fast if we know the index of the desired data.

Basic Operations -

Following are the basic primary operations of a hash table.

- **Search** − Searches an element in a hash table.
- **Insert** − inserts an element in a hash table.
- **Delete** − Deletes an element from a hash table.

Hashing - Hashing is a technique or process of mapping keys, values into the hash table by using a hash function. It is done for faster access to elements. The efficiency of mapping depends on the efficiency of the hash function used.



PROS -

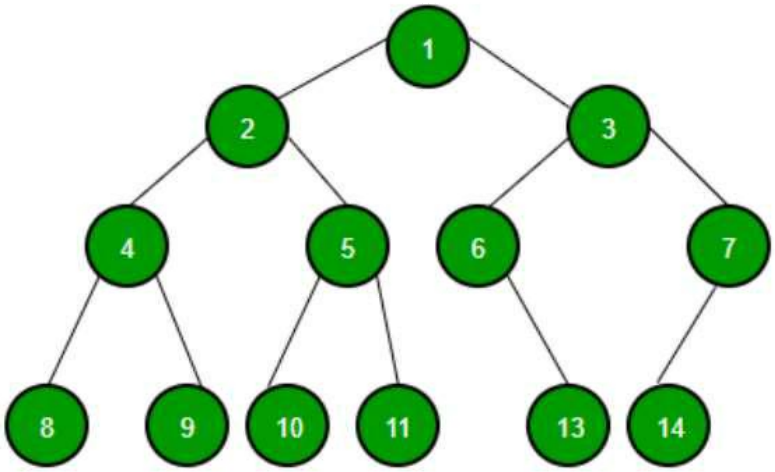
- Main advantage is synchronization.
- In many situations, hash tables turn out to be more efficient than search trees or any other table lookup structure. For this reason, they are widely used in many kinds of computer software's, particularly for associative arrays, database indexing, caches and sets.

CONS -

- Hash collisions are practically unavoidable when hashing a random subset of a large set.
- Not used as a prefix.

2) Tree –

- A tree data structure is a non-linear data structure because it does not store in a sequential manner.
- It is a hierarchical structure as elements in a Tree are arranged in multiple levels.
- In the Tree data structure, the topmost node is known as a root node.
- Each node contains some data, and data can be of any type.
- A tree whose elements have at most 2 children is called a binary tree. Since each element in a binary tree can have only 2 children, we typically name them the left and right child.
- A Binary Tree node contains the following :
    - Data
    - Pointer to left child
    - Pointer to right child



PROS -

- Solves the prefix problem (e.g, terms starting with “hyp”)

CONS -

- Slower - O (log M) [and this requires a balanced tree]

- Rebalancing binary trees is expensive.
- B-trees mitigate the rebalancing problem.

WildCard Queries

Wildcard queries are used in any of the following situations.

- The user is uncertain of the spelling of a query term (e.g., Sydney vs. Sidney, which leads to the wildcard query S*dney);
- The user is aware of multiple variants of spelling a term and (consciously) seeks documents containing any of the variants (e.g., color vs. colour);
- The user seeks documents containing variants of a term that would be caught by stemming, but is unsure whether the search engine performs stemming (e.g., judicial vs. judiciary, leading to the wildcard query judicia*);
- The user is uncertain of the correct rendition of a foreign word or phrase (e.g., the query Universit* Stuttgart).
- A query such as mon* is known as a trailing wildcard query , because the * symbol occurs only once, at the end of the search string.
- A query such as * mon is known as leading wildcard query , because the * symbol occurs only once, at the starting of the search string.

To implement Dictionary Data Structure -

How can we enumerate all terms meeting the wildcard query pro*cent ?
>> Use the forward part for "pro*", and the backward part for "*cent", then intersect them.
But this technique is so expensive.

Solution > transform wild-card queries so that the *'s always occur at the end.
This gives rise to the Permuterm Index.
Rotate query wild-card to the right.

Permuterm Index

- The Permuterm index [Garfield 1976] is a time-efficient and elegant solution to the string dictionary problem in which pattern queries may possibly include one wild-card symbol (calledTolerant Retrieval problem).
- Unfortunately Permuterm index is space inefficient because it quadruples dictionary size.

Permuterm Index Processing -

- For term ***hello***, index under:
  - ***hello$, ello$h, llohe, lohel, o$hell and $hello***

  where **$** is a special symbol.
- Queries:
  - **X** lookup on **X$** **X*** lookup on **$X***
  - ***X** lookup on **X$*** ***X*** lookup on **X***
  - **X*Y** lookup on **Y$X***

Bigram or K-gram Index

- In the Permuterm index, there are a number of lexicons for query processing. To overcome this problem K-gram technique was introduced.
- In a K-gram index , the dictionary contains all -grams that occur in any term in the vocabulary. Each postings list points from a - gram to all vocabulary terms containing that -gram.

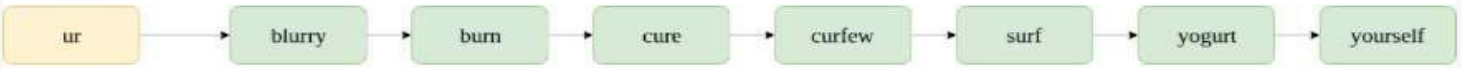
K-gram Index

- K-grams are k-length subsequences of a string. Here, k can be 1, 2, 3 and so on. For k=1, each resulting subsequence is called a “unigram”; for k=2, a “bigram”; and for k=3, a “trigram”.
- These are the most widely used k-grams for spelling correction, but the value of k really depends on the situation and context.

➔ Unigrams: [“c”, “a”, “t”, “a”, “s”, “t”, “r”, “o”, “p”, “h”, “i”, “c”]
➔ Bigrams: [“ca”, “at”, “ta”, “as”, “st”, “tr”, “ro”, “op”, “ph”, “hi”, “ic”]
➔ Trigrams: [“cat”, “ata”, “tas”, “ast”, “str”, “tro”, “rop”, “oph”, “phi”, “hic”]

<u>K-gram Index Working</u> -

- A k-gram index maps a k-gram to a postings list of all possible vocabulary terms that contain it. The figure below shows the k-gram postings list corresponding to the bigram “ur”.



Edit Distance

- In computational linguistics and computer science, edit distance is a way of quantifying how dissimilar two strings (e.g., words) are to one another by counting the minimum number of operations required to transform one string into the other.

Applications -

- Natural language processing, where automatic spelling correction can determine candidate corrections for a misspelled word by selecting words from a dictionary that have a low distance to the word in question.
- In bioinformatics, it can be used to quantify the similarity of DNA sequences, which can be viewed as strings of the letters A, C, G and T.

Spell Correction

Two principal uses -

- Correcting document(s) being indexed.
- Correcting user queries to retrieve “right” answers.

Two main flavors:

1) **Isolated word -**
    - Check each word on its own for misspelling
    - Will not catch typos resulting in correctly spelled words
      e.g., from → form

2) **Context-sensitive -**
   - Look at surrounding words
   - e.g. I flew form Heathrow to Narita.
   - Will not catch typos resulting in correctly spelled words
     e.g., from → form

Isolated Word Correction

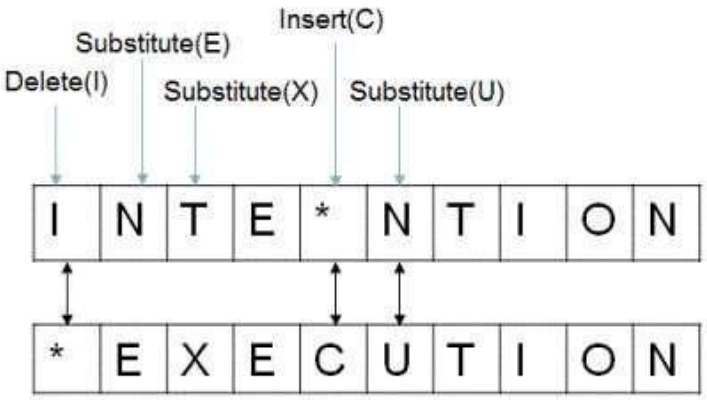
Edit Distance -

- Given two strings str1 and str2 and below operations that can be performed on str1. Find the minimum number of edits (operations) required to convert 'str1' into 'str2'.
  1) Insert 2) Remove 3) Replace
- All of the above operations are of equal cost.

<u>Example</u> - Find out the number of editing between two strings.

S1 = INTENTION

S2 = EXECUTION



-	0	E	X	E	C	U	T	I	O	N
0	0	1	2	3	4	5	6	7	8	9
I	1	1	2	3	4	5	6	6	7	8
N	2	2	2	3	4	5	6	7	7	7
T	3	3	3	3	4	5	5	6	7	8
E	4	3	4	3	4	5	6	6	7	8
N	5	4	4	4	4	5	6	7	7	7
T	6	5	5	5	5	5	5	6	7	8
I	7	6	6	6	6	6	6	5	6	7
O	8	7	7	7	7	7	7	6	5	6
N	9	8	8	8	8	8	8	7	6	5

- A phonetic algorithm is an algorithm for indexing words by their pronunciation.
- Most phonetic algorithms were developed for English and are not useful for indexing words in other languages.
- It is mainly used to correct phonetic misspellings in proper nouns.
- Algorithms for such phonetic hashing are commonly collectively known as soundex algorithms

Soundex Algorithm

Soundex Algorithm -

1) Retain the first letter of the term.
2) Change all occurrences of the following letters to '0' (zero): 'A', E', 'I', 'O', 'U', 'H', 'W', 'Y'.
3) Change letters to digits as follows:
    - B, F, P, V to 1.
    - C, G, J, K, Q, S, X, Z to 2.
    - D,T to 3.
    - L to 4.
    - M, N to 5.
    - R to 6.
4) Repeatedly remove one out of each pair of consecutive identical digits.
5) Remove all zeros from the resulting string. Pad the resulting string with trailing zeros and return the first four positions, which will consist of a letter followed by three digits.

Example -

A, E, I, O, U, H, W, Y = 0
B, F, P, V = 1
C, G, J, K, Q, S, X, Z = 2
D, T = 3
L = 4
M, N = 5
R = 6

- "Robert" maps to **R163**
- "Rupert" maps to **R163**
- "Rubin" maps to **R150**
- "Aschcraft" maps to **A2613**
- "Ashcroft" maps to **A2613**
- "Tymczak" maps to **T522**
- "Pfister" maps to **P1236**
- "Zombie" maps to **Z510**
- "Hermann" maps to **H655**

...