Problem 1 (What comes first). Consider a repository of 1000 documents of which 28 are relevant to a user query. There are two search engines A and B. Search engine A returns 10 documents and search engine B returns 20 documents. An expert says that 7 of the 10 documents returned by A are relevant while 5 of the 20 returned by B are relevant.

(a) Calculate the precision, recall and accuracy metrics of the search engines.

(b) Which search engine is the winner in terms of its i) accuracy and ii) recall?

(c) Accuracy is not the best metric for IR evaluation. Briefly justify this statement based on the previous results.

(d) Let the order of relevant documents in the corpus be 1 (most relevant), 2, ..., 28 (least relevant). We mark the documents retrieved by seach engines A and B, by relevance and **X** if not relevant and get the following results.

A: 1 **X** **X** 4 2 **X** 3 6 7 5

B: 1 3 2 4 5 **X** .. (13 irrelevant documents) .. **X**

Which search engine do you think is a better performer for the top 5 results. Why?

Problem 2 (Et tu, Brute?). Assume that you have created an inverted index of Shakespeare's Collected Works. The terms *Brutus*, *Ceasar* and *Calpurnia* have posting lists of sizes x, y and z, respectively.

a What is the time complexity for the querying the keywords `"Brutus"` AND `"Calipurnia"`? When is this achieved?

b What is the time complexity for querying `"Brutus"` OR `"Calpurnia"`?

c Make the following queries on Google (keeping uppercases intact) and report the number of estimated search results.

   (a) Caesar
   (b) Brutus
   (c) Caesar AND Brutus
   (d) Caesar OR Brutus
   (e) Brutus AND Caesar

(i) Do the numbers in the first four queries follow Boolean logic? Briefly justify.

(ii) Do the numbers in the third and fifth query follow Boolean logic? Briefly discuss, with an example, why word order should affect search results.

Problem 3 (Indexing and query processing).

(a) Create an inverted index for the corpus in Table 1. Include documents ids and frequencies(no need for positions). [1]

[1] i) Disregard stop words. ii) Stem words using the Porter Stemmer`https://text-processing.com/demo/stem/`

(b) Process the following 2 queries using TAAT (term-at-a-time) query processing and return top-1 result:
Q1: apple stores in paris.
Q2: apple songs.

i Are both answers relevant to the queries? What's the problem?

ii Briefly, propose a solution.

document id	**text**
d_1	.....With an apple I will astonish Paris.....
d_2	...I am still a very big fan of Fiona Apple....
d_3	....I love making apple strudel.....
d_4	....Apple has done a great job on the IPod....

Table 1

Albert is making sweets for his class's Christmas party. To impress his peers, he wants to search for the best recipes. As we all have sweet tooth and trust Albert's input (query), we want to help him find the best recipes by ranking them according to his query.

Problem 4 (BM25).

DocID	Document
d_9	life is a piece of cake with coffee
d_{10}	best chrismast coffee cake with nuts is this coffee cake with nuts recipe
d_{11}	sour cream coffee cake
d_{12}	carrotcake and wul nuts

Table 2

Using Okapi BM25, rank the documents in the table 2 by following the steps below. Albert's query is *coffee cake with nuts*

(a) Calculate the BM25 scores for documents and write the acquired ranking. Use the given $IDF(t)$ from the table 3 (Note that the IDFs are calculated from a big text corpora for generalization). Use $k_1 = 1.5$ and $b = 0.75$.

$$\text{score}(d, q) = \sum_{t \in q} \frac{(k_1 + 1)tf_{t,d}}{k_1\left((1-b) + b\frac{|d|}{\text{avgdl}}\right) + tf_{t,d}} \cdot \text{IDF}(t)$$

(b) This time, first remove the stop words={a, and, best, is, of, this, with} from the documents (and the query) and redo the steps in part a.

(c) What is the effect of stopword removing? (**briefly** explain)

Problem 5 (SLM).
Suppose we want to search the following collection of christmas cookie recipes. Assume that the numbers in the table indicate raw term frequencies.

Term	IDF
coffee	3.6
cake	1.9
nuts	-0.8
with	2.4

Table 3

	milk	coffee	ginger	sugar	raisins	cinnamon	flour	eggs	wulnut	apples
d_7	2	0	1	0	0	0	0	0	1	1
d_4	1	0	2	1	1	0	2	1	2	0
d_8	0	0	0	2	3	1	0	4	0	0
d_3	3	0	0	2	0	0	0	2	1	0
d_1	4	0	0	4	0	1	1	0	0	0
d_2	1	1	0	2	0	0	0	0	1	0
d_6	1	0	0	0	0	0	1	1	0	2
d_5	2	1	1	0	2	0	5	2	0	2

(a) Determine the top-3 documents including their query likelihoods for the queries

$$q_1 = \langle \text{ginger, sugar, raisins} \rangle \quad q_2 = \langle \text{milk, eggs, wulnut} \rangle$$

using the multinomial model (i.e., $P(q|d) = \prod_{t \in q} P(t|d)$) with MLE probabilities $P(t|d)$.

(b) Determine the top-3 documents using Jelinek-Mercer smoothing ($\lambda = 0.5$).

(c) Determine the top-3 documents using Dirichlet smoothing (for a suitable α)

Problem 6 (Evaluate).
Finally we want to evaluate different ranking approaches (to help Albert choose the best method).

Consider documents assessed by experts (0 = irrelevant, 1 = somewhat relevant, 2 = very relevant) in table 4, where $d_i ... d_j$ mean documents i, i+1, i+2, ..., j. Assume we used two different ranking methods to rank the documents for two different queries. And this is their results:
Ranking1(query1)= d_6, d_1, d_{11}, d_{12}, d_2, ...
Ranking2(query2)= d_{20}, d_9, d_{10}, d_3, d_4, ...
**Pay attention that the grades for query1 and query2 are different for each document.

DocId	$d_1 ... d_5$	$d_6 ... d_{10}$	$d_{11} ... d_{20}$
Grade for query1	2	1	0
Grade for query2	1	2	0

Table 4

(a) Calculate Precision@3 and Precision@5 for two ranking methods. Assume that grades 1 and 2 are mapped to "Relevant" and grade 0 is mapped to "Irrelevant".

(b) Calculate DCG@3 and DCG@5 for both ranking methods.

(c) Calculate NDCG@3 and NDCG@5.