✖

Week 2 Quiz

[redacted]

1
point

1。
Let w_1, w_2, and w_3 represent three words in the dictionary of an inverted index. Suppose we have the following document frequency distribution:

Word	Document Frequency
w_1	1000
w_2	100
w_3	10

Assume that each posting entry of document ID and term frequency takes exactly the same disk space. Which word, if removed from the inverted index, will save the **most** disk space?

● w_1

○ w_2

○ w_3

○ We cannot tell from the given information.

1
point

2。

Assume we have the same scenario as in Question 1. If we enter the query Q= “w_1 w_2” then the **minimum** possible number of accumulators needed to score all the matching documents is:

○ 1100

● 1000

○ 10

○ 100

1
point

3。
The gamma code for the term frequency of a certain document is **1110010**. What is the term frequency of the document?

○ 12

○ 11

○ 9

● 10

1
point

4。
When using an inverted index for scoring documents for queries, a shorter query always uses fewer score accumulators than a longer query.

○ True

● False

1
point

5。
What is the advantage of tokenization (normalize and stemming) before index?

☑ Reduces the number of terms (size of vocabulary)

☑ Improves performance by mapping words with similar meanings into the same indexing term

☑ Extracts words as lexical units from strings of text

1
point

6。
What can't an inverted index alone do for fast search?

○ Search document contains "A" and "B"

● Search document contains "A" or "B"

○ Retrieve documents that are relevant to the query

1
point

7。
If Zipf's law does not hold, will an inverted index be much faster or slower?

○ Faster

● Slower

1
point

8。

In BM25, the TF after transformation has upper bound

○ k

● k +1

○ 1

1
point

9。

Which of the following are weighing heuristics for the vector space model?

☑ TF weighting and transformation

☑ Document length normalization

☑ IDF weighting

1
point

10。

Which of the following integer compression has equal-length coding?

● Binary

○ γ -code

○ Unary

1
point

11。

Consider the following retrieval formula:

$$score(Q, D) = \sum_{w \in Q,D} \frac{\log(c(w, D) + 1)}{1 + \frac{avdl}{dl}} \log \frac{df(w)}{N + 1}$$

Where c(w, D) is the count of word w in document D,

dl is the document length,

avdl is the average document length of the collection,

N is the total number of documents in the collection,

and df (w) is the number of documents containing word w.

In view of TF, IDF weighting, and document length normalization, which part is missing or does not work appropriately?

○ TF

● IDF

○ Document length normalization

1
point

12。
Suppose we compute the term vector for a baseball sports news article in a collection of general news articles using **TF-IDF weighting**. Which of the following words do you expect to have the highest weight in this case?

● baseball

○ computer

○ the

[redacted] [redacted]

[redacted]

👍 👎 ⚑