This assignment was completed in stages.

Stage 1: Building Inverted Index

Here for each document, I traversed through all the words in it. The information regarding the words traversed was stored in a data structure which had the following format:

```
{
{word1 --> (docId, count), (docId,count) ...},
{word2 --> (docId, count), (docId,count) ...}
}
```

There were a few cases that I needed to consider while traversing through words
case1: If the word was not found before.
case2: If the same word was found more than once in the same document.
case3: If the same word was found more than once but in different documents.

I also maintained a dict for storing (docId, noOfWordsInDocuments), which is used in calculating the value of K in the second half of the assignment.

I stored the index in the file in the following format:

```
# word
docId count docId count
```

This made my job of building the index for the second part of the assignment easier.

Stage 2: Calculating BM25 Score

I rebuilt the index again for this part from index.out. After the index was built I calculated the BM25 score using the following formula:

$$\log \left((N - ni + 0.5) / (ni + 0.5) \right) * \{ ((k1 + 1) * fi) * ((k2 + 1) * qfi) / (K + fi) * (k2 + qfi) \}$$

I updated the formula from the original one, since we know that R and r = 0
Here,
N = no of documents
ni = no of documents which contain term i
k1 = 1.2
fi = no of time term i occurs in the document we are considering
k2 = 100

qfi = no of times term 1 occurs in the query

K = k1 * ((1 - b) + (b * float(dl) / avdl))

b = 0.75

dl = length of document, obtainable from the second data structure mainted as per the first part of the assignment.

avdl = average length of the documents in the corpus.