שאילתות באינטרנט

פרויקט חלק ראשון

תאריך הגשה: כ״ז בכסלו, 25 בדצמבר

1. תיאור השלב הראשון של הפרויקט

בשלב ראשון הנכם נדרשים לממש את ה indexer.

ה indexer הוא הרכיב במערכת הבונה את האינדקס המהופך ומאפשר את השימוש בו.

על ה indexer לתמוך בבניית אינדקס לכמויות גדולות של מסמכים ולכן אין להניח כי יש מספיק זיכרון כדי לאחסן את כל הנתונים בתוכו בבת אחת. במקום זאת, עליכם להשתמש באלגוריתמים הנלמדים בהרצאות המאפשרים את יצירת האינדקס למרות גודלו של האוסף.

בנוסף, ה indexer צריך לתמוך בעיבוד שאילתות בצורה יעילה ולכן עליכם לאחסן על הדיסק את הנתונים הרלוונטיים בצורה דחוסה. לשם כך עליכם להשתמש בשיטות הדחיסה שנלמדו בהרצאה.

ה indexer מורכב מהמחלקות הבאות:

- IndexWriter - בהינתן אוסף של מסמכים, בונה ממנו את האינדקס המהופך על הדיסק בצורה יעילה מבחינת זמן ומקום.
- IndexReader – מאפשר גישה אל האינדקס המהופך על מנת לקבל נתונים שונים.

1.1. אוסף המסמכים

ה indexer יקבל את אוסף המסמכים כקובץ קלט אחד המכיל את כל המסמכים הקיימים באוסף. בין מסמך למסמך מפרידה שורה של 80 כוכביות (*).

דוגמאות לקבצים עם כמויות שונות של מסמכים ניתן למצוא כאן

מכיוון שלמסמכים שבקובץ אין מזהה (ID), יש למספר את המסמכים בסדר עולה. כלומר המסמך הראשון יקבל את המזהה 1, המסמך השני את המזהה 2 וכן הלאה.

לצורך זיהוי המילים השונות של האוסף יש לבצע את הפעולות הבאות:

- חלוקת הטקסט למילים נפרדות בכל מקום שבו יש תו שאינו אלפאנומרי (אינו אות או ספרה). התווים שאינם אלפאנומריים צריכים להיות מושלכים.
- נרמול הטקסט על ידי הפיכת כל תווי האותיות לאותיות קטנות (lowercase).

1.2. תיאור התרגיל

בהינתן קובץ הקלט עם המסמכים, עליכם ליצור את האינדקס. האינדקס צריך להיות מאוחסן על הדיסק עם ההגבלות הבאות:

- אסור להשתמש במערכת של מסד נתונים כדי לשמור את המידע. עליכם לממש את האחסון בעצמכם.
- ניתן להשתמש ביותר מקובץ אחד כדי לאחסן את האינדקס. אולם, מספר הקבצים שייווצרו צריך להיות קבוע ולא תלוי במספר המסמכים או גודל המילון וכדומה.

1.3. דרישות הקוד

התכנית תכיל לפחות את שתי המחלקות הבאות: (ככל הנראה התכנית תכלול מחלקות נוספות הנחוצות לצורך מימושן)

1.3.1. IndexWriter

בהינתן קובץ קלט, המחלקה תיצור אינדקס על הדיסק שאפשר יהיה לגשת אליו מאוחר יותר. כל הנתונים שישתמשו בהם מאוחר יותר צריכים להיות מאוחסנים באינדקס שעל הדיסק.

המטרה היא לבנות אינדקס יעיל הן מבחינת הזמן הדרוש לבנייתו והן מבחינת גודלו על הדיסק.

בנוסף, מאפשרת המחלקה למחוק את האינדקס מהדיסק על ידי מחיקת כל הקבצים מספריית האינדקס.

```
class IndexWriter:
    def __init__(self, inputFile, dir):
```

```
        """Given a collection of documents, creates an
        on disk index
        inputFile is the path to the file containing
        the review data (the path includes the filename
        itself)
        dir is the name of the directory in which all
        index files will be created
        if the directory does not exist, it should be
        created"""

    def removeIndex(self, dir):
        """Delete all index files by removing the given
        directory
        dir is the name of the directory in which all
        index files are located. After removing the
        files, the directory should be deleted."""
```

IndexReader .1.3.2

לאחר שנוצר אינדקס על הדיסק, ניתן להשתמש במתודות של המחלקה כדי לגשת לנתונים הקיימים באינדקס. ניתן להניח כי המתודות יופעלו רק לאחר שהאינדקס ייבנה על ידי ה IndexWriter.

```
class IndexReader
    def __init__(self, dir):
        """Creates an IndexReader which will read from
        the given directory
        dir is the name of the directory in which all
        index files are located."""

    def getTokenFrequency(self, token):
        """Return the number of documents containing a
        given token (i.e., word)
        Returns 0 if there are no documents containing
        this token"""

    def getTokenCollectionFrequency(self, token):
        """Return the number of times that a given
        token (i.e., word) appears in the whole
        collection.
        Returns 0 if there are no documents containing
        this token"""

    def getDocsWithToken(self, token):
```

```
        """Returns a series of integers of the form id-
        1, freq-1, id-2, freq-2, ... such
        that id-n is the n-th document containing the
        given token and freq-n is the
        number of times that the token appears in doc
        id-n
        Note that the integers should be sorted by id.
        Returns an empty Tuple if there are no
        documents containing this token"""


    def getNumberOfDocuments(self):
        """Return the number of documents in the
        collection"""
```

1.4. הגשת התרגיל

- התרגיל יוגש דרך אתר המכללה בפורמט ZIP
- עבור כל זוג סטודנטים יש להגיש רק הגשה אחת. שם הקובץ צריך להיות ID1_ID2.zip כאשר ID1 ו ID2 הם מספרי הזהות של הסטודנטים המגישים את הפרויקט.
- קבצי הקוד צריכים לכלול שני קבצים עם השמות IndexWriter.py ו IndexReader.py. תכנית הבדיקה תייבא (import) קבצים אלו כך שחשוב שתקפידו על השמות הנכונים (כולל אותיות גדולות וקטנות). קבצים אלו יכילו את מימוש המחלקות אותן נדרשתם לממש.
- במידה והפרויקט שלכם מכיל קבצים נוספים (מחלקות נוספות), באחריותכם לייבא אותם (import) מתוך הקבצים IndexWriter.py ו IndexReader.py.
- יש לוודא כי הקובץ שהועלה הוא בפורמט הנכון וכולל את כל קבצי הקוד הנחוצים להרצת הפרויקט.

1.5. בדיקת התרגיל

בבדיקת התרגיל ייעשה שימוש בקבצי הנתונים המצויים בקישור שלמעלה או בקבצים אחרים בפורמט זהה.

בדיקת התרגיל נעשית בעזרת מערכת אוטומטית. כדי שבדיקת התרגיל לא תיכשל (ותגרום להורדה בציון) הקפידו היטב על ההנחיות שבסעיף הקודם.