Movie Review Sentiment Analysis (CS2)

This assignment is designed to give you experience with hash tables, and is based off of a programming competition question (https://www.kaggle.com/c/sentiment-analysis-on-movie-reviews) regarding sentiment analysis and machine learning.

> Sentiment Analysis: the process of computationally identifying and categorizing opinions expressed in a piece of text, especially in order to determine whether the writer's attitude towards a particular topic, product, etc., is positive, negative, or neutral.
>
> Machine Learning: a branch of computer science that explores the construction and study of algorithms that can learn from data. Such algorithms operate by building a model from example inputs and using that to make predictions or decisions.

The data that the algorithm is going to "learn" from is a set of 8,529 movie reviews in which the sentiment of each review has been manually rated on a scale from 0 to 4. The sentiment labels are:

0 - negative
1 - somewhat negative
2 - neutral
3 - somewhat positive
4 - positive

The data has been formatted so it is easy for C++ programs to identify each word (or punctuation). The data looks like this:

```
1 A series of escapades demonstrating the adage that what is good for the goc
4 This quiet , introspective and entertaining independent is worth seeking .
1 Even fans of Ismail Merchant 's work , I suspect , would have a hard time s
3 A positively thrilling combination of ethnography and all the intrigue , be
1 Aggressive self-glorification and a manipulative whitewash .
4 A comedy-drama of nearly epic proportions rooted in a sincere performance b
1 Narratively , Trouble Every Day is a plodding mess .
3 The Importance of Being Earnest , so thick with wit it plays like a reading
1 But it does n't leave you with much .
1 You could hate it for the same reason .
1 There 's little to recommend Snow Dogs , unless one considers cliched dialc
1 Kung Pow is Oedekerk 's realization of his childhood dream to be in a marti
4 The performances are an absolute joy .
3 Fresnadillo has something serious to say about the ways in which extravagar
3 I still like Moonlight Mile , better judgment be damned .
3 A welcome relief from baseball movies that try too hard to be mythic , this
3 a bilingual charmer , just like the woman who inspired it
2 Like a less dizzily gorgeous companion to Mr. Wong 's In the Mood for Love
1 As inept as big-screen remakes of The Avengers and The Wild Wild West .
2 It 's everything you 'd expect -- but nothing more .
```

The assignment is to use the provided data to develop an algorithm that will allow a user to input a new review and will automatically score the sentiment of the review.

The program will require that you

1. Read in a review
2. Assign each word in the review the score attributed to the review
3. Enter a WordEntry object (consisting of the word, total score, and number of

occurrences) into a hash table. if word already exists in the hash table, update the score and number of occurrences to the record

4. Repeat Step 1 until all data is entered

The main.cpp program that does this is provided for you. Your responsibility is to implement the HashTable.cpp and WordEntry.cpp files. The corresponding .h files are supplied on the course website.

Basic Assignment (30 points). The program should prompt the user to input a movie review, and automatically score the review based on the average score of the words in the review. The program must implement all methods in the WordEntry.cpp and HashTable.cpp files correctly.

Example output:

enter a review -- Press return to exit:
A weak script that ends with a quick and boring finale
The review has an average value of 1.79128
Negative Sentiment

enter a review -- Press return to exit:
Loved every minute of it
The review has an average value of 2.39219
Positive Sentiment

Suggested Additional Options:
Improved Assignment (10 points). While this algorithm works fairly well, experiment with your solution and discover instances where the sentiment is not predicted well. Discuss how you could improve the performance of the algorithm and implement it. Compare your updated algorithm's accuracy to the basic assignment. Give examples. *Possible suggestions include identifying words that the learning algorithm has seemed to have slightly skewed. An updated algorithm could identify these words with more accurate results.*

Class Template (10 points). The hash table class used for the basic assignment will only work with WordEntry objects. Create a Class Template that would allow your hash table class to be used with any type of object.

Extremes (2 points). Identify the most negative and most positive word.

Most Often Occurring (2 points). Identify the most often occurring words.