CMP2003 Data Structures and Algorithms (C++)
Term Project
Instructor : Assistant Prof. Tevfik Aytekin

--Frequent Words in Tweets --

1. Main Requirements

You are expected to write a c++ console application which reads a text file that consists of tweets and then print the top 10 most frequent words in the tweets. In order to get the tweet data file, first download the following zip file:

http://cs.stanford.edu/people/alecmgo/trainingandtestdata.zip

When you unzip the file you will find the tweet data file which is named:

“training.1600000.processed.noemoticon.csv”

The format of the training data

The data is a CSV with emoticons removed. Data file format has 6 fields:

0 - the polarity of the tweet (0 = negative, 2 = neutral, 4 = positive)
1 - the id of the tweet
2 - the date of the tweet
3 - the query. If there is no query, then this value is NO_QUERY.
4 - the user that tweeted.
5 - the text of the tweet.

Note that the tweet text is in the last field.

Stopwords

You can find a list of stopwords here:

https://www.link-assistant.com/seo-stop-words.html

You should not count these words.

Definition of a word

For simplicity assume that any contiguous block of alphabetic characters (letters from “a” to “z”, both upper and lower case) which includes at most one single quotation mark between these letters is a word. According to this definition the following tweet:

"@dweeman why aren't you a happy camper?"

has the words:

dweeman
why
aren't
you
a
happy
camper

After reading and processing is over, your program must list the "top 10" most frequent words in the tweets as follows:

Sample output :

```
<word1>      <word count>
<word2>      <word count>
<word3>      <word count>
<word4>      <word count>
<word5>      <word count>.
.
.
.
<word10>      <word count>

Total Elapsed Time : X seconds
```

Whole application can be implemented with console facilities (you do not need advanced GUI elements). The project consists of two parts.

A. Implementation of data structure.

This will be a proper C++ class. You must be able to create many instances of this class.
(You should not use third party libraries including C++ STL, Boost etc.). However you can use, **iostream**, **ctime**, **fstream**, **string** like IO and string related classes.

B. The main program itself. Your program must calculate the elapsed time starting from reading tweet file to the end of the printing top 10 words. There is no need to ask user input.

2. Submission

You are expected to submit

a. A working copy of your program executable
b. The Visual Studio project directory (config files) and source code
c. A small report on what you did: description of the data structures and algorithms you used, sample execution and how to build the application and run. Even better, use a screen capture program like Camtasia Studio, to record a sample run of your executable in video, to make our life easy.

The project is at most **3 PERSON** size.
We encourage everybody to work in groups.

The deadline is set 28 December 2018 11:59 pm.

Submit your files from itslearning system.

Demos will be scheduled and announced later.
Late submissions will get lower grade by 10% for each day.

3. Cheating Policy.

You are not supposed to use each other’s source code. Also you should not use source code from internet, another person or your book’s examples.

All the source codes will be filtered through a similarity analysis tool, which is known to be effective against many types of code copying and changing tricks. These projects will be graded as 0.

4. Evaluation

60%: Correctness of program, data structures and algorithms used.
30%: Running time. We will sort all projects according to their running times and distribute these points (30%) according to this sorted list. The top one will get all the points and the last one will get no points.
(10%): Project report.

Any lack of the items mentioned above (source code with visual studio config files, executable and project report), crashing executables and existence of viruses may cause you to get very low grades.

In the demonstrations all the group members must be present. All the members will be asked questions and a common grade will be given to all group members. Therefore wrong or inconsistent answers will affect all group member’s grade.

5. Bonuses

You can get bonuses for extra efforts :

* Good coding styles and OO programming skills

* Making the sort function parameterizable with a generic comparison class. (Hence you can sort your list according to any criterion)

* Or any other nice feature you can think of.

Please mention such extra efforts..