

The Manhattan Project

7

Generated by Doxygen 1.8.13

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	Collection Class Reference	5
3.1.1	Constructor & Destructor Documentation	5
3.1.1.1	Collection()	6
3.1.2	Member Function Documentation	6
3.1.2.1	addDocument()	6
3.1.2.2	addToIDF()	6
3.1.2.3	compareSimilar()	6
3.1.2.4	printSimilarTbl()	6
3.1.2.5	sim()	6
3.1.2.6	TFIDF()	7
3.1.3	Member Data Documentation	7
3.1.3.1	Docs	7
3.1.3.2	IDF	7
3.1.3.3	library	7
3.1.3.4	Novern	7
3.1.3.5	NumDocs	7
3.1.3.6	similarities	7

3.2	ParseBox Class Reference	8
3.2.1	Constructor & Destructor Documentation	8
3.2.1.1	ParseBox()	8
3.2.2	Member Function Documentation	8
3.2.2.1	isAlphaNum()	8
3.2.2.2	isLegal()	8
3.2.2.3	longestLegalIndex()	8
3.2.2.4	propCap()	9
3.2.2.5	puncOnlyEOS()	9
3.2.2.6	purePuncCheck()	9
3.3	Replace Class Reference	9
3.3.1	Member Function Documentation	9
3.3.1.1	addCapital()	9
3.3.1.2	addExceptions()	10
3.3.1.3	contains()	10
3.3.1.4	swapString()	10
3.4	Stemmer Class Reference	10
3.4.1	Constructor & Destructor Documentation	10
3.4.1.1	Stemmer()	10
3.4.2	Member Function Documentation	10
3.4.2.1	getStem()	11
3.4.2.2	shouldStem()	11
3.5	Word Class Reference	11
3.5.1	Constructor & Destructor Documentation	11
3.5.1.1	Word()	11
3.5.2	Member Function Documentation	11
3.5.2.1	getCount()	12
3.5.2.2	getWord()	12
3.5.2.3	incrementWord()	12
3.5.2.4	setWord()	12

3.6	WordCounts Class Reference	12
3.6.1	Constructor & Destructor Documentation	13
3.6.1.1	WordCounts()	13
3.6.2	Member Function Documentation	13
3.6.2.1	add()	13
3.6.2.2	capitalFix()	13
3.6.2.3	checkCaps()	13
3.6.2.4	CLindex()	14
3.6.2.5	countWords()	14
3.6.2.6	getFirstLegalAlphaIndex()	14
3.6.2.7	getFirstLegalPuncIndex()	14
3.6.2.8	implicitSentence()	14
3.6.2.9	isAlphaChar()	14
3.6.2.10	isAlphanum()	15
3.6.2.11	L()	15
3.6.2.12	printList()	15
3.6.2.13	propCap()	15
3.6.2.14	puncOnlyEOS()	15
3.6.2.15	removeCapAmb()	15
3.6.2.16	S()	15
3.6.2.17	scanFileIn()	16
3.6.2.18	sortList()	16
3.6.2.19	split()	16
3.6.2.20	splitAdd()	16
3.6.2.21	stringLength()	16
3.6.2.22	updateLevelInfo()	16
3.6.3	Member Data Documentation	16
3.6.3.1	addTmp	16
3.6.3.2	capChecks	17
3.6.3.3	capTmp	17
3.6.3.4	exceptionTable	17
3.6.3.5	letterCounter	17
3.6.3.6	regBox	17
3.6.3.7	sentenceCounter	17
3.6.3.8	stemBox	17
3.6.3.9	stringList	17
3.6.3.10	tmp	18
3.6.3.11	wordCount	18
3.6.3.12	wordCounter	18
3.6.3.13	wordList	18

4 File Documentation	19
4.1 Collection.cpp File Reference	19
4.2 Collection.h File Reference	19
4.3 main.cpp File Reference	19
4.3.1 Function Documentation	20
4.3.1.1 exceptionFail()	20
4.3.1.2 fileFail()	20
4.3.1.3 main()	20
4.3.1.4 runCollection()	20
4.3.1.5 runReadScore()	20
4.3.1.6 Usage()	21
4.4 ParseBox.cpp File Reference	21
4.5 ParseBox.h File Reference	21
4.6 Replace.cpp File Reference	21
4.7 Replace.h File Reference	21
4.8 Stemmer.cpp File Reference	22
4.9 Stemmer.h File Reference	22
4.10 Word.cpp File Reference	22
4.11 Word.h File Reference	22
4.12 WordCounts.cpp File Reference	22
4.13 WordCounts.h File Reference	23
Index	25

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Collection	5
ParseBox	8
Replace	9
Stemmer	10
Word	11
WordCounts	12

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Collection.cpp	19
Collection.h	19
main.cpp	19
ParseBox.cpp	21
ParseBox.h	21
Replace.cpp	21
Replace.h	21
Stemmer.cpp	22
Stemmer.h	22
Word.cpp	22
Word.h	22
WordCounts.cpp	22
WordCounts.h	23

Chapter 3

Class Documentation

3.1 Collection Class Reference

```
#include <Collection.h>
```

Public Member Functions

- [Collection](#) (int N)
- int [addDocument](#) (istream &fileName, [Replace](#) exceptions)
- bool [compareSimilar](#) ()
- void [printSimilarTbl](#) ()
- double [sim](#) ([WordCounts](#) &docA, [WordCounts](#) &docB)
- double [TFIDF](#) ([Word](#) &wordIn)

Protected Member Functions

- void [addToIDF](#) (const [WordCounts](#) &words)

Protected Attributes

- int [NumDocs](#)
- double [Novern](#)
- int [Docs](#)
- map< string, int > [IDF](#)
- vector< [WordCounts](#) > [library](#)
- vector< vector< double > > [similarities](#)

3.1.1 Constructor & Destructor Documentation

3.1.1.1 Collection()

```
Collection::Collection (
    int N = 0 )
```

3.1.2 Member Function Documentation

3.1.2.1 addDocument()

```
int Collection::addDocument (
    istream & fileName,
    Replace exceptions )
```

3.1.2.2 addToIDF()

```
void Collection::addToIDF (
    const WordCounts & words ) [protected]
```

3.1.2.3 compareSimilar()

```
bool Collection::compareSimilar ( )
```

3.1.2.4 printSimilarTbl()

```
void Collection::printSimilarTbl ( )
```

3.1.2.5 sim()

```
double Collection::sim (
    WordCounts & docA,
    WordCounts & docB )
```

3.1.2.6 TFIDF()

```
double Collection::TFIDF (
    Word & wordIn )
```

3.1.3 Member Data Documentation

3.1.3.1 Docs

```
int Collection::Docs [protected]
```

3.1.3.2 IDF

```
map<string, int> Collection::IDF [protected]
```

3.1.3.3 library

```
vector<WordCounts> Collection::library [protected]
```

3.1.3.4 Novern

```
double Collection::Novern [protected]
```

3.1.3.5 NumDocs

```
int Collection::NumDocs [protected]
```

3.1.3.6 similarities

```
vector<vector<double> > Collection::similarities [protected]
```

The documentation for this class was generated from the following files:

- [Collection.h](#)
- [Collection.cpp](#)

3.2 ParseBox Class Reference

```
#include <ParseBox.h>
```

Public Member Functions

- [ParseBox](#) ()
- bool [isLegal](#) (const string &word)
- int [longestLegalIndex](#) (const string &word)
- bool [propCap](#) (const string &word)
- bool [puncOnlyEOS](#) (const string &word)
- bool [isAlphaNum](#) (const string &word)
- bool [purePuncCheck](#) (const string &word)

3.2.1 Constructor & Destructor Documentation

3.2.1.1 ParseBox()

```
ParseBox::ParseBox ( )
```

3.2.2 Member Function Documentation

3.2.2.1 isAlphaNum()

```
bool ParseBox::isAlphaNum (
    const string & word )
```

3.2.2.2 isLegal()

```
bool ParseBox::isLegal (
    const string & word )
```

3.2.2.3 longestLegalIndex()

```
int ParseBox::longestLegalIndex (
    const string & word )
```

3.2.2.4 propCap()

```
bool ParseBox::propCap (
    const string & word )
```

3.2.2.5 puncOnlyEOS()

```
bool ParseBox::puncOnlyEOS (
    const string & word )
```

3.2.2.6 purePuncCheck()

```
bool ParseBox::purePuncCheck (
    const string & word )
```

The documentation for this class was generated from the following files:

- [ParseBox.h](#)
- [ParseBox.cpp](#)

3.3 Replace Class Reference

```
#include <Replace.h>
```

Public Member Functions

- bool [addExceptions](#) (istream &inExc)
- string [swapString](#) (const string &in)
- void [addCapital](#) (const string &cap)
- bool [contains](#) (const string &cap)

3.3.1 Member Function Documentation

3.3.1.1 addCapital()

```
void Replace::addCapital (
    const string & cap )
```

3.3.1.2 addExceptions()

```
bool Replace::addExceptions (
    istream & inExc )
```

3.3.1.3 contains()

```
bool Replace::contains (
    const string & cap )
```

3.3.1.4 swapString()

```
string Replace::swapString (
    const string & in )
```

The documentation for this class was generated from the following files:

- [Replace.h](#)
- [Replace.cpp](#)

3.4 Stemmer Class Reference

```
#include <Stemmer.h>
```

Public Member Functions

- [Stemmer](#) ()
- bool [shouldStem](#) (const string &word)
- void [getStem](#) (string &word)

3.4.1 Constructor & Destructor Documentation

3.4.1.1 Stemmer()

```
Stemmer::Stemmer ( )
```

3.4.2 Member Function Documentation

3.4.2.1 `getStem()`

```
void Stemmer::getStem (
    string & word )
```

3.4.2.2 `shouldStem()`

```
bool Stemmer::shouldStem (
    const string & word )
```

The documentation for this class was generated from the following files:

- [Stemmer.h](#)
- [Stemmer.cpp](#)

3.5 Word Class Reference

```
#include <Word.h>
```

Public Member Functions

- [Word](#) (string wordIn="")
- string [getWord](#) () const
- int [getCount](#) ()
- void [incrementWord](#) ()
- void [setWord](#) (string newWord)

3.5.1 Constructor & Destructor Documentation

3.5.1.1 `Word()`

```
Word::Word (
    string wordIn = "" ) [inline]
```

3.5.2 Member Function Documentation

3.5.2.1 getCount()

```
int Word::getCount ( ) [inline]
```

3.5.2.2 getWord()

```
string Word::getWord ( ) const [inline]
```

3.5.2.3 incrementWord()

```
void Word::incrementWord ( ) [inline]
```

3.5.2.4 setWord()

```
void Word::setWord (
    string newWord ) [inline]
```

The documentation for this class was generated from the following file:

- [Word.h](#)

3.6 WordCounts Class Reference

```
#include <WordCounts.h>
```

Public Member Functions

- [WordCounts](#) (const [Replace](#) &exec)
- int [scanFileIn](#) (istream &inFile)
- void [sortList](#) ()
- void [printList](#) ()
- void [countWords](#) ()
- int [stringLength](#) ()
- void [updateLevelInfo](#) (const string &word)
- double [L](#) ()
- double [S](#) ()
- double [CLindex](#) ()
- void [implicitSentence](#) ()

Adds one to sentence if there is an implicit Sentence in the file.
- void [splitAdd](#) (string &word)
- void [add](#) (string word)
- void [split](#) (string word)
- bool [isAlphanum](#) (string word)
- int [getFirstLegalPuncIndex](#) (string word)
- int [getFirstLegalAlphIndex](#) (string word)
- bool [isAlphaChar](#) (char c)
- bool [propCap](#) (string word)
- bool [puncOnlyEOS](#) (string word)
- void [capitalFix](#) ()
- void [removeCapAmb](#) ()
- void [checkCaps](#) ()

Public Attributes

- vector< [Word](#) > [wordList](#)
- vector< string > [stringList](#)
- vector< int > [capChecks](#)
- int [wordCount](#)
- [ParseBox](#) [regBox](#)
- [Stemmer](#) [stemBox](#)
- [Replace](#) [exceptionTable](#)
- string [tmp](#)
- string [addTmp](#)
- string [capTmp](#)
- int [letterCounter](#)
- int [wordCounter](#)
- int [sentenceCounter](#)

3.6.1 Constructor & Destructor Documentation

3.6.1.1 WordCounts()

```
WordCounts::WordCounts (
    const Replace & exec ) [inline]
```

3.6.2 Member Function Documentation

3.6.2.1 add()

```
void WordCounts::add (
    string word )
```

3.6.2.2 capitalFix()

```
void WordCounts::capitalFix ( )
```

3.6.2.3 checkCaps()

```
void WordCounts::checkCaps ( )
```

3.6.2.4 CLindex()

```
double WordCounts::CLindex ( )
```

3.6.2.5 countWords()

```
void WordCounts::countWords ( )
```

3.6.2.6 getFirstLegalAlphIndex()

```
int WordCounts::getFirstLegalAlphIndex (
    string word )
```

3.6.2.7 getFirstLegalPuncIndex()

```
int WordCounts::getFirstLegalPuncIndex (
    string word )
```

3.6.2.8 implicitSentence()

```
void WordCounts::implicitSentence ( )
```

Adds one to sentence if there is an implicit Sentence in the file.

Note

if the file ends in end of sentence token then nothing is changed

3.6.2.9 isAlphaChar()

```
bool WordCounts::isAlphaChar (
    char c )
```

3.6.2.10 isAlphanum()

```
bool WordCounts::isAlphanum (
    string word )
```

3.6.2.11 L()

```
double WordCounts::L ( )
```

3.6.2.12 printList()

```
void WordCounts::printList ( )
```

3.6.2.13 propCap()

```
bool WordCounts::propCap (
    string word )
```

3.6.2.14 puncOnlyEOS()

```
bool WordCounts::puncOnlyEOS (
    string word )
```

3.6.2.15 removeCapAmb()

```
void WordCounts::removeCapAmb ( )
```

3.6.2.16 S()

```
double WordCounts::S ( )
```

3.6.2.17 scanFileIn()

```
int WordCounts::scanFileIn (
    istream & inFile )
```

3.6.2.18 sortList()

```
void WordCounts::sortList ( )
```

3.6.2.19 split()

```
void WordCounts::split (
    string word )
```

3.6.2.20 splitAdd()

```
void WordCounts::splitAdd (
    string & word )
```

3.6.2.21 stringLength()

```
int WordCounts::stringLength ( )
```

3.6.2.22 updateLevelInfo()

```
void WordCounts::updateLevelInfo (
    const string & word )
```

3.6.3 Member Data Documentation

3.6.3.1 addTmp

```
string WordCounts::addTmp
```

3.6.3.2 capChecks

```
vector<int> WordCounts::capChecks
```

3.6.3.3 capTmp

```
string WordCounts::capTmp
```

3.6.3.4 exceptionTable

```
Replace WordCounts::exceptionTable
```

3.6.3.5 letterCounter

```
int WordCounts::letterCounter
```

3.6.3.6 regBox

```
ParseBox WordCounts::regBox
```

3.6.3.7 sentenceCounter

```
int WordCounts::sentenceCounter
```

3.6.3.8 stemBox

```
Stemmer WordCounts::stemBox
```

3.6.3.9 stringList

```
vector<string> WordCounts::stringList
```

3.6.3.10 tmp

```
string WordCounts::tmp
```

3.6.3.11 wordCount

```
int WordCounts::wordCount
```

3.6.3.12 wordCounter

```
int WordCounts::wordCounter
```

3.6.3.13 wordList

```
vector<Word> WordCounts::wordList
```

The documentation for this class was generated from the following files:

- [WordCounts.h](#)
- [WordCounts.cpp](#)

Chapter 4

File Documentation

4.1 Collection.cpp File Reference

```
#include <Collection.h>
#include <math.h>
#include <iostream>
```

4.2 Collection.h File Reference

```
#include <Collection.h>
#include <Replace.h>
#include <WordCounts.h>
#include <map>
```

Classes

- class [Collection](#)

4.3 main.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <WordCounts.h>
#include <Replace.h>
#include <Collection.h>
```

Functions

- int [Usage](#) (char *arg0)
- int [fileFail](#) (char *arg2)
- int [exceptionFail](#) (char *arg1)
- int [runCollection](#) (int argc, char *argv[])
- int [runReadScore](#) (int argc, char *argv[])
- int [main](#) (int argc, char *argv[])

4.3.1 Function Documentation

4.3.1.1 [exceptionFail\(\)](#)

```
int exceptionFail (  
    char * arg1 )
```

4.3.1.2 [fileFail\(\)](#)

```
int fileFail (  
    char * arg2 )
```

4.3.1.3 [main\(\)](#)

```
int main (  
    int argc,  
    char * argv[] )
```

4.3.1.4 [runCollection\(\)](#)

```
int runCollection (  
    int argc,  
    char * argv[] )
```

4.3.1.5 [runReadScore\(\)](#)

```
int runReadScore (  
    int argc,  
    char * argv[] )
```

4.3.1.6 Usage()

```
int Usage (
    char * arg0 )
```

4.4 ParseBox.cpp File Reference

```
#include <ParseBox.h>
#include <regex>
#include <string>
#include <stdio.h>
#include <iostream>
```

4.5 ParseBox.h File Reference

```
#include <regex>
#include <string>
#include <stdio.h>
#include <iostream>
```

Classes

- class [ParseBox](#)

4.6 Replace.cpp File Reference

```
#include <Replace.h>
```

4.7 Replace.h File Reference

```
#include <iostream>
#include <fstream>
#include <map>
#include <set>
#include <regex>
```

Classes

- class [Replace](#)

4.8 Stemmer.cpp File Reference

```
#include <Stemmer.h>
#include <iostream>
```

4.9 Stemmer.h File Reference

```
#include <regex>
#include <string>
```

Classes

- class [Stemmer](#)

4.10 Word.cpp File Reference

```
#include <Word.h>
```

4.11 Word.h File Reference

```
#include <string>
```

Classes

- class [Word](#)

4.12 WordCounts.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <vector>
#include <string>
#include <algorithm>
#include <WordCounts.h>
#include <regex>
```

4.13 WordCounts.h File Reference

```
#include <iostream>
#include <fstream>
#include <vector>
#include <algorithm>
#include <string>
#include <Word.h>
#include <regex>
#include <ParseBox.h>
#include <Stemmer.h>
#include <Replace.h>
```

Classes

- class [WordCounts](#)

Index

- add
 - WordCounts, [13](#)
- addCapital
 - Replace, [9](#)
- addDocument
 - Collection, [6](#)
- addExceptions
 - Replace, [9](#)
- addTmp
 - WordCounts, [16](#)
- addToIDF
 - Collection, [6](#)
- CLIndex
 - WordCounts, [13](#)
- capChecks
 - WordCounts, [16](#)
- capTmp
 - WordCounts, [17](#)
- capitalFix
 - WordCounts, [13](#)
- checkCaps
 - WordCounts, [13](#)
- Collection, [5](#)
 - addDocument, [6](#)
 - addToIDF, [6](#)
 - Collection, [5](#)
 - compareSimilar, [6](#)
 - Docs, [7](#)
 - IDF, [7](#)
 - library, [7](#)
 - Novern, [7](#)
 - NumDocs, [7](#)
 - printSimilarTbl, [6](#)
 - sim, [6](#)
 - similarities, [7](#)
 - TFIDF, [6](#)
- Collection.cpp, [19](#)
- Collection.h, [19](#)
- compareSimilar
 - Collection, [6](#)
- contains
 - Replace, [10](#)
- countWords
 - WordCounts, [14](#)
- Docs
 - Collection, [7](#)
- exceptionFail
 - main.cpp, [20](#)
- exceptionTable
 - WordCounts, [17](#)
- fileFail
 - main.cpp, [20](#)
- getCount
 - Word, [11](#)
- getFirstLegalAlphIndex
 - WordCounts, [14](#)
- getFirstLegalPuncIndex
 - WordCounts, [14](#)
- getStem
 - Stemmer, [10](#)
- getWord
 - Word, [12](#)
- IDF
 - Collection, [7](#)
- implicitSentence
 - WordCounts, [14](#)
- incrementWord
 - Word, [12](#)
- isAlphaChar
 - WordCounts, [14](#)
- isAlphaNum
 - ParseBox, [8](#)
- isAlphanum
 - WordCounts, [14](#)
- isLegal
 - ParseBox, [8](#)
- L
 - WordCounts, [15](#)
- letterCounter
 - WordCounts, [17](#)
- library
 - Collection, [7](#)
- longestLegalIndex
 - ParseBox, [8](#)
- main
 - main.cpp, [20](#)
- main.cpp, [19](#)
 - exceptionFail, [20](#)
 - fileFail, [20](#)
 - main, [20](#)
 - runCollection, [20](#)
 - runReadScore, [20](#)
 - Usage, [20](#)

- Novern
 - Collection, [7](#)
- NumDocs
 - Collection, [7](#)
- ParseBox, [8](#)
 - isAlphaNum, [8](#)
 - isLegal, [8](#)
 - longestLegalIndex, [8](#)
 - ParseBox, [8](#)
 - propCap, [8](#)
 - puncOnlyEOS, [9](#)
 - purePuncCheck, [9](#)
- ParseBox.cpp, [21](#)
- ParseBox.h, [21](#)
- printList
 - WordCounts, [15](#)
- printSimilarTbl
 - Collection, [6](#)
- propCap
 - ParseBox, [8](#)
 - WordCounts, [15](#)
- puncOnlyEOS
 - ParseBox, [9](#)
 - WordCounts, [15](#)
- purePuncCheck
 - ParseBox, [9](#)
- regBox
 - WordCounts, [17](#)
- removeCapAmb
 - WordCounts, [15](#)
- Replace, [9](#)
 - addCapital, [9](#)
 - addExceptions, [9](#)
 - contains, [10](#)
 - swapString, [10](#)
- Replace.cpp, [21](#)
- Replace.h, [21](#)
- runCollection
 - main.cpp, [20](#)
- runReadScore
 - main.cpp, [20](#)
- S
 - WordCounts, [15](#)
- scanFileIn
 - WordCounts, [15](#)
- sentenceCounter
 - WordCounts, [17](#)
- setWord
 - Word, [12](#)
- shouldStem
 - Stemmer, [11](#)
- sim
 - Collection, [6](#)
- similarities
 - Collection, [7](#)
- sortList
 - WordCounts, [16](#)
- split
 - WordCounts, [16](#)
- splitAdd
 - WordCounts, [16](#)
- stemBox
 - WordCounts, [17](#)
- Stemmer, [10](#)
 - getStem, [10](#)
 - shouldStem, [11](#)
 - Stemmer, [10](#)
- Stemmer.cpp, [22](#)
- Stemmer.h, [22](#)
- stringLength
 - WordCounts, [16](#)
- stringList
 - WordCounts, [17](#)
- swapString
 - Replace, [10](#)
- TFIDF
 - Collection, [6](#)
- tmp
 - WordCounts, [17](#)
- updateLevelInfo
 - WordCounts, [16](#)
- Usage
 - main.cpp, [20](#)
- Word, [11](#)
 - getCount, [11](#)
 - getWord, [12](#)
 - incrementWord, [12](#)
 - setWord, [12](#)
 - Word, [11](#)
- Word.cpp, [22](#)
- Word.h, [22](#)
- wordCount
 - WordCounts, [18](#)
- wordCounter
 - WordCounts, [18](#)
- WordCounts, [12](#)
 - add, [13](#)
 - addTmp, [16](#)
 - CLIndex, [13](#)
 - capChecks, [16](#)
 - capTmp, [17](#)
 - capitalFix, [13](#)
 - checkCaps, [13](#)
 - countWords, [14](#)
 - exceptionTable, [17](#)
 - getFirstLegalAlphaIndex, [14](#)
 - getFirstLegalPuncIndex, [14](#)
 - implicitSentence, [14](#)
 - isAlphaChar, [14](#)
 - isAlphanumeric, [14](#)
 - L, [15](#)
 - letterCounter, [17](#)

- printList, [15](#)
- propCap, [15](#)
- puncOnlyEOS, [15](#)
- regBox, [17](#)
- removeCapAmb, [15](#)
- S, [15](#)
- scanFileIn, [15](#)
- sentenceCounter, [17](#)
- sortList, [16](#)
- split, [16](#)
- splitAdd, [16](#)
- stemBox, [17](#)
- stringLength, [16](#)
- stringList, [17](#)
- tmp, [17](#)
- updateLevelInfo, [16](#)
- wordCount, [18](#)
- wordCounter, [18](#)
- WordCounts, [13](#)
- wordList, [18](#)
- WordCounts.cpp, [22](#)
- WordCounts.h, [23](#)
- wordList
 - WordCounts, [18](#)