Advanced Object Oriented System Design Using C++

Assignment 2

Overview

This is a straight-forward assignment to read in a number of words from standard input in to a vector<string> and output a subset of those words.

Task

The task for this assignment is:

1. Read in a list of words (whitespace-separated std::string values) from **std::cin** (until EOF or an error) and put each word in a (single!) **std::vector<string>**.
2. Use the **std::find()** algorithm to find the iterator, **pos1**, of the first instance of the word "begin".
3. Use the **std::find()** algorithm to find the iterator, **pos2**, of the first instance of the word "end" after the first instance of the word "begin".
4. Output all of the words in the words **between pos1 and pos2** using **std::copy**. (*pos1, i.e., the word "begin", must not be output.) Each word output must be followed by a space character.

Tips

- Reading in whitespace-separated words is easy: by default C++ will skip over whitespace when using operator >>. This means cin >> str; where str is a std::string variable, will read in whitespace-separated words. :-)
- After calling std::find() the first time, check if pos1 != v.end() and if it is not, then ++pos1. This will move pos1 to one position after the "begin" position, or, do nothing if "begin" was not found!
- std::copy's third argument can be ostream_iterator<string>(cout, " ").

Sample Input Files

```
$ cat a2-input1.dat
bad output1
begin
  a
  quick brown
  fox
  jumped
end
bad output2
$ cat a2-input2.dat
bad output1
begin
  a
  quick brown
  fox
  jumped
okay output2
$ cat a2-input3.dat
bad output1
  a
  quick brown
  fox
  jumped
end
```

```
bad output2
```

Sample Program Runs

```
 $ g++-10.2.0 -std=<a href="https://moodle.cs.uwindsor.ca/mod/glossary/showentry.php?
eid=7&displayformat=dictionary" title="C++-Related Glossary: C++20" class="glossary autolink
concept glossaryid1">c++20</a> -Wall -Wextra -Werror a2-soln.cxx
 $ ./a.out <a2-input1.dat
 a quick brown fox jumped
 $ ./a.out <a2-input2.dat
 a quick brown fox jumped okay output2
 $ ./a.out <a2-input3.dat

 $
```

Submission

Remember to upload your C++ program code to this page when done! :-)

Submission status

Attempt number	This is attempt 1.
Submission status	Submitted for grading
Grading status	Not marked
Due date	Monday, 22 February 2021, 11:59 PM
Time remaining	4 days 15 hours

Grading criteria

This rubric is a general rubric for compiled programming languages.

Normally the marker will adhere to the rubric, but, the marker reserves the right to add/subtract marks based on the quality of the work provided. (Such overrides will be noted in the feedback comments.)

Compiler Warnings	Too many compiler warnings. ***0 points***	No more than two (2) compiler warnings. ***1 points***	No compiler warnings. ***2 points***
Compiler Errors (Deduction)	Code fails to compile due to compiler errors. ***-5 points***	No compiler errors. ***0 points***	
Possible Run-Time Errors	Too many run-time errors occur or can occur. ***0 points***	Only a few minor run-time errors occur or can occur. ***1 points***	No run-time errors occur or can occur. ***2 points***
Code Structure, Names, and Comments	Poor code structure, meaningful symbol names, and comments. ***0 points***	Satisfactory code structure, meaningful symbol names, and comments. ***1 points***	
Packaging	Not all files are provided and/or not all targets are built cleanly and completely. ***0 points***	All files are provided. All targets are built cleanly and completely. ***1 points***	
Unit Tests	Too many tests (provided or not) don't pass having input or output processing and correctness issues. ***0 points***	Nearly all tests (provided or not) pass and have no input or output processing and correctness issues. ***1 points***	All tests (provided or not) pass and have no input or output processing and correctness issues. ***2 points***
Assignment Requirements Conformance	Not all instructions were followed. ***0 points***	Most instructions were followed. ***1 points***	All instructions were followed. ***2 points***
Late Penalty (Deduction)	Late. All marks deducted. ***-10 points***	Late but submitted within 24 hours of deadline or late without penalty deadline. ***-2 points***	Not late or late without penalty (as decided by or with permission of instructor). ***0 points***

Last modified Thursday, 18 February 2021, 8:31 AM

File submissions A2.cxx 18 February 2021, 8:31 AM

Submission comments Comments (0)

Edit submission | Remove submission

You can still make changes to your submission.

◄ Programming Paradigms

Jump to...

Slides: From Pointers to Iterators ►