UPE Tutoring:

CS 31 Project 5 Hack

Sign-in: https://goo.gl/enMfW1

Slides link available upon sign-in

Overview

- Two functions
  - makeProper
    - Put patterns into proper form
  - rate
    - Find rating of document

Key Requirements

Definitions

- *Document*: Text of web page
- *Pattern*: Two words (w1 and w2) and a nonnegative integer separation
- *Match*: Document matches pattern if somewhere, w2 appears before or after w1 with no more than separation words in between
- *Proper Form*: Form for collection of patterns, meeting requirements in spec
- *Rating of Document*: Number of patterns it matches

Two Functions

- Proper form
  - Transform upper case letters into lower case
  - Remove patterns for which
    - A word contains no characters or non-letter characters
    - Negative separation
  - For two patterns with same w1 and w2 in either order, remove the pattern with lower separation value
- Rate
  - Return number of patterns document matches

Example Walk-Through

- W1: "mad", "deranged", "nefarious", "have"
  W2: "scientist", "robot",   "plot",   "mad"
  Separation: 1, 3, 0, 12

- Document: "The mad UCLA scientist unleashed a deranged evil giant robot."
  - Rating: 2
- Document: "deranged deranged robot deranged robot robot"
  - Rating: 1
- Document: " That plot: NEFARIOUS!"
  - Rating: 1

Tips & Strategies

Library Functions You May Find Helpful

```
// Compares 2 strings;
// Returns: < 0 if str1 < str2
//           0 if str1 == str2
//          > 0 if str1 > str2
int strcmp ( const char* str1, const char* str2 );


// Returns length of C-string not including null byte;
// i.e. strlen("cat") returns 3
size_t strlen ( const char* str );
```

Library Functions You May Find Helpful

```
// Copies the string from source to destination
// Note: make sure the source string fits in
// destination, otherwise undefined behavior!!!
char* strcpy ( char* destination, const char* source );
```

Tips

- How do I start?
  - Work on one function first, develop incrementally
    - Create smaller helper functions and get small parts working at a time
  - If you want, create a separate project working with regular strings as a starting point
    - Change it to C Strings little-by-little

Tips

- How do I "delete" extra items at the end of an array?
  - Hint: does it matter?
- What size limits can I assume for the array?
  - Detailed in specification
- A C string is an array of characters followed by the zero byte
  - Keep this in mind when working with arrays of C strings!

Tips

- Test thoroughly to avoid losing points
  - Assert statements to consider all cases
- Reread specification to verify requirements
- Beware of common errors with C strings
  - Errors involving zero byte, size, etc.
- Visual C++ users, use #define _CRT_SECURE_NO_WARNINGS before your include statements

Good luck!

Sign-in   https://goo.gl/enMfW1
Slides   https://goo.gl/J7Mqtm
Practice   https://github.com/uclaupe-tutoring/practice-problems/wiki

Questions? Need more help?
- Come up and ask us! We'll try our best.
- UPE offers daily computer science tutoring:
  - Location: ACM/UPE Clubhouse (Boelter 2763)
  - Schedule: https://upe.seas.ucla.edu/tutoring/
- You can also post on the Facebook event page.