University of New Brunswick, Faculty of Computer Science

CS 2263 Lab 4 Winter 2018

Due Date: Friday, March 2, 12 pm.

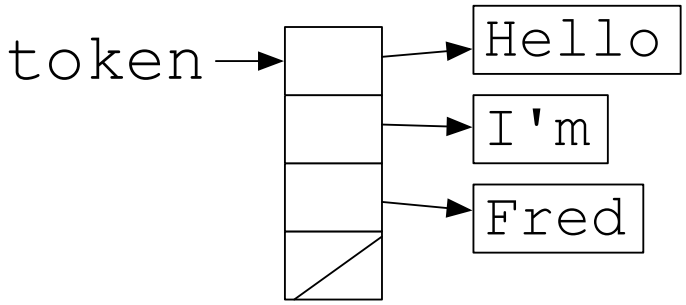
Purpose: Practice with arrays of strings, `malloc`/`free`, header files and make-files.

Tokenize and Reverse

Your task is to implement the following functions in one file, and test them from a `main` function in another file.

```
char **stringToTokens(char *str, char *sep);
```

This function tokenizes `str`, using the separator characters specified in `sep`. It puts deep copies of the tokens into a dynamically allocated array of strings, and returns the array. This array includes an extra element with the value of `NULL`, which allows looping through the array without knowing its length. For example, given the string `"Hello, I'm Fred"`, and the separator `", "`, the following array will be created:



Note that it has four elements, with the final element being `NULL`.

```
void destroyTokens(char **tokenArray);
```

This function deallocates the memory allocated for an array of strings, as created by `stringToTokens`. Be sure to free each string in addition to the array of pointers.

```
void reverse(char *s);
```

This function reverses string `s` in place.

Header File

Write a header file that includes the prototypes for the above three functions, as well as a descriptive comment for each function. If you have included other helper functions in the implementation file, don't include them in the header file.

Main Function

Write a C program in a separate file to test these functions. This program prompts the user for a sentence, calls `stringToTokens` to tokenize the sentence into an array of strings, then reverses and prints each string in the array using `reverse`, and finally frees the array using `destroyTokens`. For example:

```
$ ./testTokens
enter sentence:
Hello, I'm Fred
olleH m'I derF
```

Makefile

Include a `makefile` to build your program, with commands appropriate for the FCS Linux system. You will be penalized if your program is not correctly compiled and linked using your makefile.

To pass in the lab: Create a single pdf document with listings of the two `.c` files, the header file, and the `makefile`, and a record of a terminal session. Also include a zip file containing the source files, header file, and `makefile`. Submit these files to the Desire2Learn dropbox. Name your documents `LastName_FirstName_Lab4.pdf` and `LastName_FirstName_Lab4.zip` (`LastName` and `FirstName` are of course substituted with your last and first name).