CS221 - Lab03

Paul Haskell

INTRODUCTION

In this lab you will use some of the helpful functions from the C library to write programs that are slightly more complex than from the previous lab.

Programs for this Lab

Your first program is called **doubles.c** . This program reads zero or more `double` values from the command line arguments. Each `double` value should be cast to a `short` and then should be printed out.

Easy…but how do we convert the strings from the command line to `doubles`? We can use the `atof()` method. Be sure to

```
#include <stdlib.h>
```

to get the declaration for this method.

The next program is **tolower.c** . This program reads zero or more strings from the command line, converts each to lowercase, and prints out the results.

This would be really easy in Java or Python, huh? But C does not have a `tolower()` method that converts strings to lowercase. You get to write that yourself. How should that method work?

- what does it do with input characters that are digits or punctuation?
- what does it do with input characters that are already lowercase?
- what does it do with uppercase characters?

The third program **shuffle.c** is a bit more fun. Read a single integer from the first command line argument. The `atoi()` method will help you get the integer value. Create an array of integers whose length is the number you read from the command line (it will be > 0). Fill the array with the integer values 0 through N-1.

Now you will shuffle the array. You will need some way to make random numbers. Now C is an old language, and its libraries contain several different attempts to do this well. We will use the original, called `rand()` . Please

```
#include <stdlib.h>
```

to get access to the functions in this library.

To initialize the library, you first must call the `srand()` function. This function needs a `long` input value, which should be some number of your choosing. (In a week or two we will learn how to use the current time as an ever-changing argument to `srand()`). Now you can get a random integer between 0 and 2 billion by calling the `rand()` method. This method doesn't need any input arguments. You should figure out how to use `rand()` to shuffle your array.

Once you've shuffled the array, please print it out.

CONCLUSION:

Please push the three programs to GitHub before the lab deadline. The deadline for **doubles.c** is the end of class. The other two programs are due by 11:59pm Weds Feb 5th.

These programs are still pretty simple. The first big project is coming soon.

Task	Score, points
doubles.c compiles and runs	1
doubles.c gives correct output values for each of 5 test inputs	5
tolower.c compiles and runs	1
tolower.c gives correct output values for each of 5 test inputs	10
shuffle.c compiles and runs	1
shuffle.c gives correct output values for each of 5 test inputs	15