Faculty of Science, Engineering and Technology

Introduction to Programming

Tutorial Task 11.1: C File Handling

Overview

This program will allow you to demonstrate simple reading and writing from a file using C.

Purpose: Develop a simple C program that reads and writes from a file and uses loops and arrays.

Task: Create your own file reading program. Submit to Doubtfire when complete.

Time: This task should be started in the Week 11 lab class and submitted for feedback before the start of Week 12.

Resources:

- Programming Arcana
- Google
- Swinburne CodeCasts (YouTube Channel, iTunesU)
  - Learning a new language
  - Introducing Objects

Submission Details

You must submit the following files to Doubtfire:

- read_write_file.c source code
- Screenshot of the Bash Terminal showing the execution of your program.

Make sure that your task has the following in your submission:

- The program must read the required details from the file, store the read data in an array and then print out each line to the terminal.
- Code must follow coding convention and standards used in the unit (layout, and use of case).
- The code must compile and the screenshot show it working.
- Your program must have the indicated local variables, and use them appropriately.

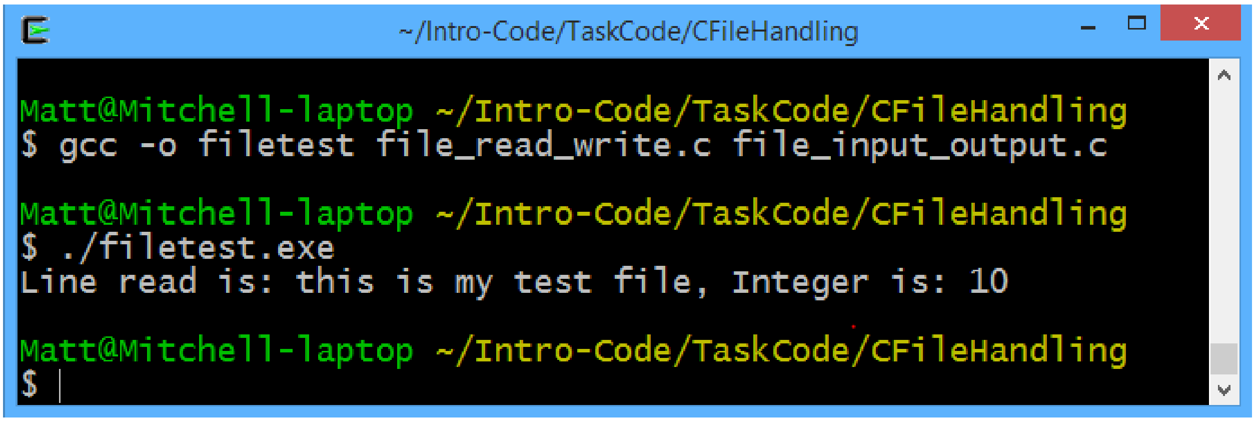
Instructions

Files allow you to store data persistently. In this task you will take a simple C file reading program and modify it to read multiple lines using a loop, store each line as an element in an array, then print the lines to the Bash terminal screen.

To explore this topic, we will create a C program that will:

- Open a file and read in the number of lines in the file.
- Loop according to the number of lines to read in each line.
- Store each line in an array.
- Print out the lines from the array to the terminal.

The screen shot below shows how to compile and run the sample code provided in the resources for this task:



```
Matt@Mitchell-laptop ~/Intro-Code/TaskCode/CFileHandling
$ gcc -o filetest file_read_write.c file_input_output.c

Matt@Mitchell-laptop ~/Intro-Code/TaskCode/CFileHandling
$ ./filetest.exe
Line read is: this is my test file, Integer is: 10

Matt@Mitchell-laptop ~/Intro-Code/TaskCode/CFileHandling
$ 
```

1. The integer read from the first line of the file should indicate the number of text lines that follow. Your program should then loop reading one line after another into an array of my_string (or char[]'s) until all the text lines have been read in. Call a procedure, passing in the array and the number of lines, that prints out to the terminal all the text lines that read from the file.

2. Comment out the write_lines() procedure. Instead you will need to create a text file (**mytestfile.dat**) with a number on the first line (to indicate the following number of text lines) and then lines of text. Use notepad or Sublime Text to create the file.

```
Procedure: Main
---------------------------------
Variables:
 - numberOfLines (to store an Integer value read from the file)
 - lineArray[20]  (to store String values read from the file)
---------------------------------
Steps:
 1: Open the file
 2. Read in the number of text lines in the file.
 3. Loop reading each line of line text into an array until all the lines have been read.
 4: Call the procedure read_lines_to_terminal(FILE *file_ptr)
 5: Print to the terminal: ‘The file contained: ‘
 6. Loop printing to the terminal each element of the array on a new line.
```

Now that the Task is complete you can submit it to Doubtfire – include a screenshot of your code compiling and running.