

Project Part 1: Introduction to C Programming Language, and File I/O

Overview

This is Part 1 of a project consisting of four parts. Part 1 of the project is worth 5% of your final course grade. Refer to your Suggested Course Schedule to confirm your Part 1 due date.

Read all project directions and questions before you begin preparing your answers and consult your Open Learning Faculty Member if you have any questions about the project.

The overall project in this course is to design and develop a simple data link layer protocol. Each part of the project will help you to successfully complete the overall project in this course. Because most network programs require a fast processing language, they use C programming languages. For this reason the C programming language is used in this project.

Objective

It is assumed that you know Java programming language, but you may not have used C programming language before. The purpose of Part 1 in this Project is to introduce and provide you with some experience and comfort in using C programming language.

Instructions

1. Read the section called *Project Part 1: Background Information*. You will need this information to complete Part 1 of the project and to understand the project computer requirements. This section describes the similarities and differences between C and Java, gives an example of code that will allow you to print messages, and explains how to manipulate files with a sample code.
2. Read all project directions before you begin preparing your answers. Feel free to consult your Open Learning Faculty Member if you have any questions about the project.
3. Write programs as follows:
 - a. Implement the following functions in a file without using strcpy(), strlen() and strcat():
 - i. void string_copy (char *destination, char *source)

- ii. `int string_length (char *s)`
 - iii. `void string_cat (char *destination, char *source)`
- b. Implement a program in another file, which:
- i. Receives a string from the user as an argument;
 - ii. Stores the characters of your student number one by one in a `char` array variable;
 - iii. Prints the above two variables and length of strings (from i and ii), using one `printf()` statement; and
 - iv. Includes examples to test the above 3 functions, with proper assignments and `printf()` statements.
4. Write a program that copies a file to another:
- i. The two file names will be given to the program as arguments from the user.
 - ii. The first file will be copied into the second file.
 - iii. You should compare the two files to see if they are the same.
 - iv. Do not use `FILE` functions. You must use `open()`, `close()`, `read()` and `write()`.

The above assignments will help you to use strings properly and to do file manipulation. The basic method of using files in C programs and networking protocols is similar. This assignment will help you to move on to the next parts of the project.

Requirements for Project Part 1

You need to submit the programs with screenshots showing how your programs are compiled and run. Please refer to the general guidelines for preparing and submitting your project found under the Assignments Overview tab of your course.

Project Marking Criteria

Each Project Part is marked according to the following criteria and weightings:

Marking Criteria	Weighting
No syntax error: All requirements are fully implemented without syntax errors. Submitted screen shots will be reviewed with source code.	/5
Correct implementation: All requirements are correctly implemented and produce correct results Submitted screen shots will be reviewed with source code.	/5
Total	/10