README

This file provides the documentation and the listing of each file and a little explanation of each and also the steps to test the code.

My goal for the semester was to write a “preprocessor” that takes a “c” file reads it and for each function defined in that file, if finds out the variables that this function depends upon. This functionality is very important for the main project that we were working on, which is to figure out the impact on the performance on the Invite system, if it knows whether it has seen the current state or not and if it has, then it skips the test. My code comes into play, when we talk about a state. We decided to differentiate among the states, based on the values for each variable, the current test depends upon. My code finds out these variables and after the integration, will pass along these variables to the next “preprocessor that creates the resulting code for testing.

I divided the code in multiple files with each file housing the code called on the similar levels. I am listing the files and the functions in each with a little explanation for each.

The List of Files and Directories:

a. The CSInvite directory

CSInvite.h → It is the main header file for all the components. It has the “includes” from the c library. It also contains the macros for the lengths and numbers. Then it defines the data structures for all the variables used by the program. And at the end, it has the prototypes of all the functions used by the program.

CSInvite.c → The main file. It calls all the other functions and does controls the program flow. It contains the main function.

CSInvite_help.c → This file holds the three string helper functions. These functions are called by other functions to do string manipulation.

CSInvite_set_input.c → This file holds the functions that are called by the main function once after the invocation of the program. It sets up the global variables and constants, and the functions with their definitions.

CSInvite_set_func.c → This file sets up the scopes of the statements, plus the re-declared variables for each function, the user asks for.

CSInvite_set_vars.c → This file contains the functions that actually fill the structure that represents the variables the function depends upon. It also contains the function to handle calls to other functions and recursion.

Makefile → makes CSInvite.exe (can be used in Windows or Linux)

b. The Test Directory

This directory contains the test files. These tests were used to check the functionality of the program.

test.h, test.c, test1.c and test2.c → These are the basic test files, test1.c and test2.c have one function definition and test.c calls these functions.

single_test.c → This file contains seven function definitions, these functions test various features the program provides.

Makefile → makes the test.exe, also checks single_test.c for problems.

NOTE: "compile" the test code before using it with the "CSInvite", for the program to work correctly.