CMPE230
PROJECT-2

Student Name: Hasan Öztürk
Student Number: 2017400258

I followed object oriented manner in order to implement the project. There are 2 classes whose fields are used in the main code.

Class1: headers

There are 2 fields in this class:
dir: it stores the directory of a .h file. Further it is used in the makefile.
name: it is the file name of a .h file. Further it is used in the makefile.

Class2: cFiles

There are 3 fields in this class
list: it is a list which contains the names of the included header files of a particular .c file. It is filled by parsing the particular .c file's content
dir: it stores the directory of a .c file. Further it is used in the makefile.
name: it is the file name of a .c file. Further it is used in the makefile.

There are 2 lists named cList and headerList which contain header and cFiles objects respectively.

The most important part of the project is handled in the while loop. This loop traverses all the subdirectories of the given directory which is taken as an argument from the terminal. Then it finds all the files ending with .c or .h and creates cFile and headers instances respectively. After that fields of the objects are initialized and put into the cList or headerList according to the condition.
In the following for loop all the .c files are opened and their included header files are detected.

WARNING CHECK:
Included header files is stored in headerNameList and all headerFiles are stored in headerList. The program checks if there is a header file that is

in headerList but not in headerNameList. If there is such an element, it generates a warning message.

ERROR CHECK:

Included header files is stored in headerNameList and all headerFiles are stored in headerList. The program checks if there is a header file that is in headerNameList but not in headerList. If there is such an element, it generates a nerror message.

WRITING THE MAKEFILE:

First written target is the “program” which creates the executable named “program”. It’s dependencies are the object files created from the .c files. Dependencies and the gcc command is written by using the objects of cFile class which is stored in cList.

Following targets are the object files which are dependent to .c files and their .h files. Dependencies and the gcc commands are also written by using the objects of cFile class which is stored in cList.

TESTING THE PROJECT:

I write the code to a file that has .py as extention. This is in the same file as testcases. I run the code in terminal with following the command:

```
./project2.py cmpe230fall2017hw2_Testcases_updated/tc5u/
```

After that makefile is generated to the same directory as project2.py. Then I write:

```
make
```

Then the executable file ‘program’ is generated and I write

```
./program
```

I see the right output for all cases. My program gave error message for tc3 and warning message for tc4.