

Project #4 [60 points]

Due date is Tuesday, April 30th.

Task:

In this project you are being asked to write a program that will read in a text file based on a command line argument and then work with that file, producing an output file.

The command line arguments that your program should take are -h, -i and -o. -i lets the person running specify the input file (default should be input.txt). -o lets the user specify the output file (default should be output.txt).

An example of running this project would be, if your executable was called palin,

```
./reverser -i data.in -o data.out
```

to read data from the file data.in and send data to data.out

The input file will consist of many lines of numbers, separated by spaces, with no line longer than 80 characters.

An example input file would be something like this:

```
5 17 20 37 8
57 17 17 5 3 9
1 2 3 4 5 6
7 8 1 837 43847
```

I want you to go through this file, line by line. Starting with the first number, read it number by number into a stack data structure that you create. This stack should be implemented using a structure, with a fixed or dynamically allocated array and an integer storing the location of the topmost element and appropriate functions that work with that structure (push, pop). An example of this structure would be something similar to:

```
struct Stack {
    int top;
    int stack[100];
};
```

Then, using that stack, output the numbers to your output file, again line by line. Treat each line separately. The result should be a file where each output line is the reverse of the input line, with one condition. **I want your stack functions to ignore duplicates.** So your stack should, before adding an element to the stack, ensure that the number is not already in the stack.

Details:

You must implement this project using at least 3 source files, reverser.c, stack.h, stack.c, You also must have a makefile that will compile the project for me. If you do not include a makefile your project will not be run.

In addition, I want you to take a screenshot of you using the debugger to examine the elements of your stack array in the middle of running your code. That is because I want you to immediately start using the debugger and so this is part of your credit.

Submission:

Submit the 3 source files, your makefile, and the proof of debugging as separate files in canvas (not zipped up please).