Benjamin Frederickson

[bfrederickson@dmacc.edu](mailto:bfrederickson@dmacc.edu)

Data Structures Final Project Report

Formalized Proposal

My project is a writing program. This program allows you to use a variety of commands to allow the user to edit the words they have typed. This will allow the user to customize their writing to their own desires and allow them to record whatever they wish. Thus, solving the real-world problem. Realistically this program is a flawed version of better word writing programs due to some limitations of knowledge, time, and tools given.

A more specific description of this program is that it uses a linked list and two stacks to do everything in the program. The linked list holds strings of what the user has typed. The two stacks hold the last piece of information that was typed, deleted, or inserted. These two data structures work together to create a writing program.

There are several commands the user can use to edit what they’ve typed into the linked list. The first one being InsertWord which allows the user to insert a word at any index value of the linked list. The second one is DeleteWord it allows the user to delete a word at any index value of the linked list. The third being UndoWord which allows the user to undo their last action, whether it was deleting a word, inserting one, undoing a redo, or just undoing what the user last typed. The fourth is RedoWord it allows the user to redo their last action, like UndoWord. RedoWord can redo a deletion, an insertion, an undo, or just redo the last word typed. The final and fifth command is EndProgram which ends the program for the user.

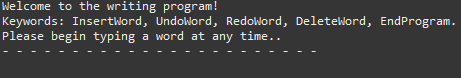
Time/Change Logs

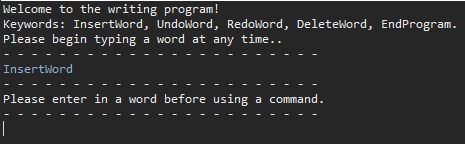
Lessons Learned

CODE :

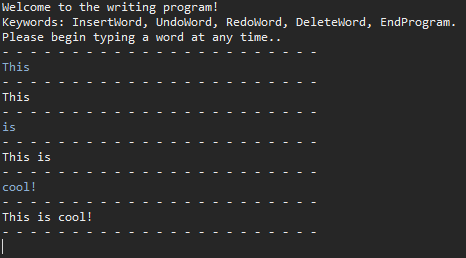
User Manual

Welcome to the user manual. In this manual you will learn how to use the writing program. It’s relatively simple to work through due to the fact the program prompts each action. We’ll start with what you see when you start the program:

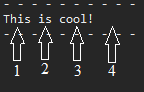
In the beginning of the program it prompts you to type a word at any time, do so and the program will begin.

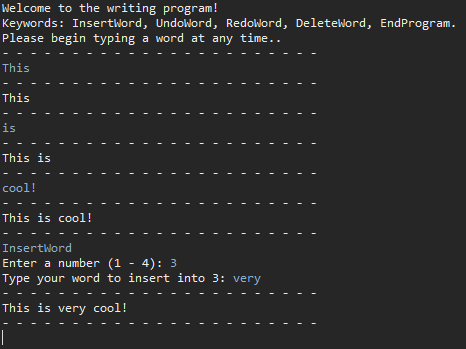
If you happen to ignore the request to type a word and enter a command (other than EndProgram), you will encounter this error:

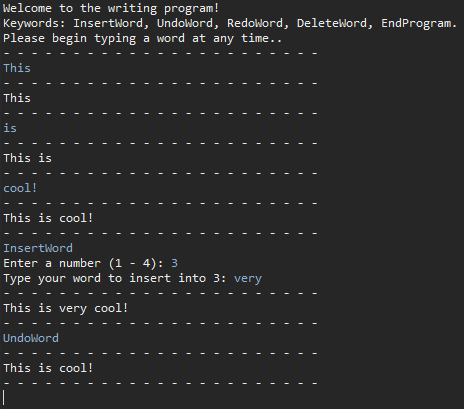
Simply just make sure to type a word (any word at all) to allow the use of commands. This program needs at least one word to allow you to do anything with commands.

Once you begin typing some words you will be able to use any commands you would like. To begin we will type, “This” then press enter to insert it into the program. We will do the same for “is”, and “cool!” to see them on screen.

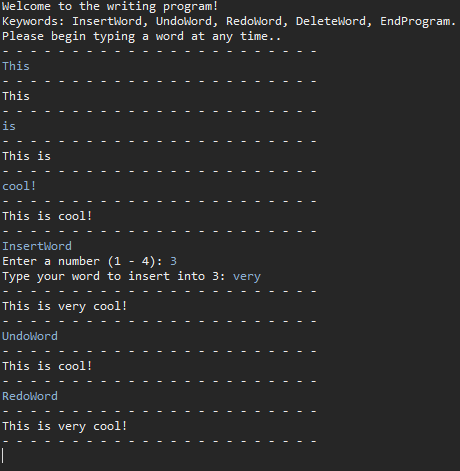
Once you’ve added these words into the program, we will begin teaching you how to use each command starting with InsertWord.

**InsertWord**: To start this command it is simple, if you type InsertWord into the program it will begin. Once you’ve done that it will prompt you to enter the number of where you want to place the word. The number values are simple. For the numbers from our example please reference this screenshot:

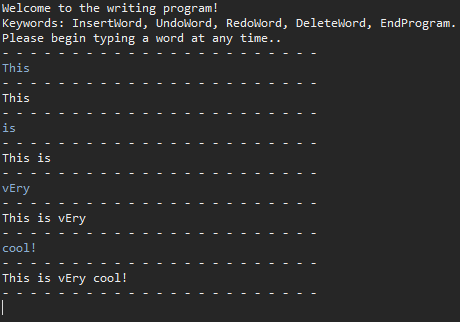
As you can see there is four choices, depending on how many words you type there will be a number value for each space. So, if you had 5 numbers, each word would be numbered from 1 to 5 with an extra number representing a 6th space. When you choose a number, your word will be placed in that number value. To show what that exactly means we will have to finish our example. For our example we will choose 3. Next the program will ask you type in a word to insert into the value you’ve selected previously. For our example we will type “very”. This is what it will look like after we have finished the InsertWord command:

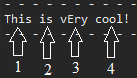
**UndoWord**: To use this command all you have to do is type UndoWord. Once you’ve done that the program will undo your last action. For example, after we just inserted a word, we could use the UndoWord command to remove it from our words typed. This screenshot will show you just that:

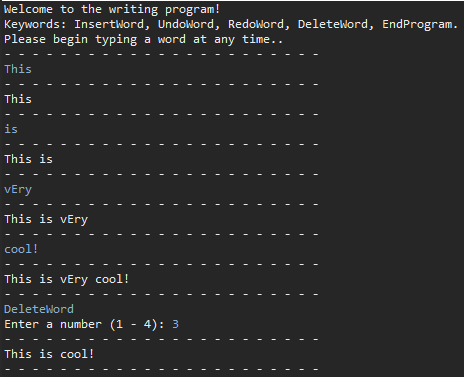
This command can also be used to undo a deletion, a word you’ve typed or undo the RedoWord command. Each of these is accomplished by just entering the UndoWord command. No other information or values needed.

**RedoWord**: To use this command you just have to type RedoWord. This will then redo your last action in the program. For example, after we just undid the last action using the UndoWord command, you can redo it and make it insert the “very” again. This screenshot will allow you to see this in action:

Like the UndoWord command, this command can redo a deletion, redo a word you just typed, or redo an insertion. You don’t need to do anything other than type in the RedoWord command, and then it will do just as the name says; redo.

**DeleteWord**: The DeleteWord command is like the InsertWord command but has some distinct differences. For the DeleteWord command we will start a new example. In this example we will use the same words “This is very cool!” but with a misspelling in the word “very”. It will look like “vEry”, thus we need to delete it. Here is a screenshot to show the example progress so far:

Like the command InsertWord, the program will prompt you for a number value to delete. So, type in the number corresponding to our example. Here is a screenshot to show what numbers correspond to our example:



For our example we would choose the number 3. After choosing the number you are done using the command and the program will delete the word. This is the main difference between DeleteWord and InsertWord. Here is a screenshot showing the finished command:

**Errors during commands**: If you encounter any errors during the commands reference these three points:

(If you encounter an error not listed please contact the IT department ASAP)

1. You entered a word instead of a number when prompted for a number.

- Enter a number and this issue is solved.

2. You entered a number too high or low when prompted for a range.

- Each command will give you a range of numbers and if you don’t follow then you will encounter an error. Just choose a number in the range and this issue is solved.

3. You entered UndoWord twice.

- This isn’t allowed you need to choose a different command or type a word.

Conclusion