Indira Gandhi Delhi Technical University for Women

(Established by Govt. of Delhi vide Act 09 of 2012)

Kashmere Gate, Delhi - 110006

B.Tech. CSAI

DATA STRUCTURES USING C

Subject Code: BCS 201

STACK BASED TEXT EDITOR

PROJECT FILE

***Submitted to*:**

Dr. Shweta Singhal

***Submitted by*:**

03001172020 - Jhanvee Khola

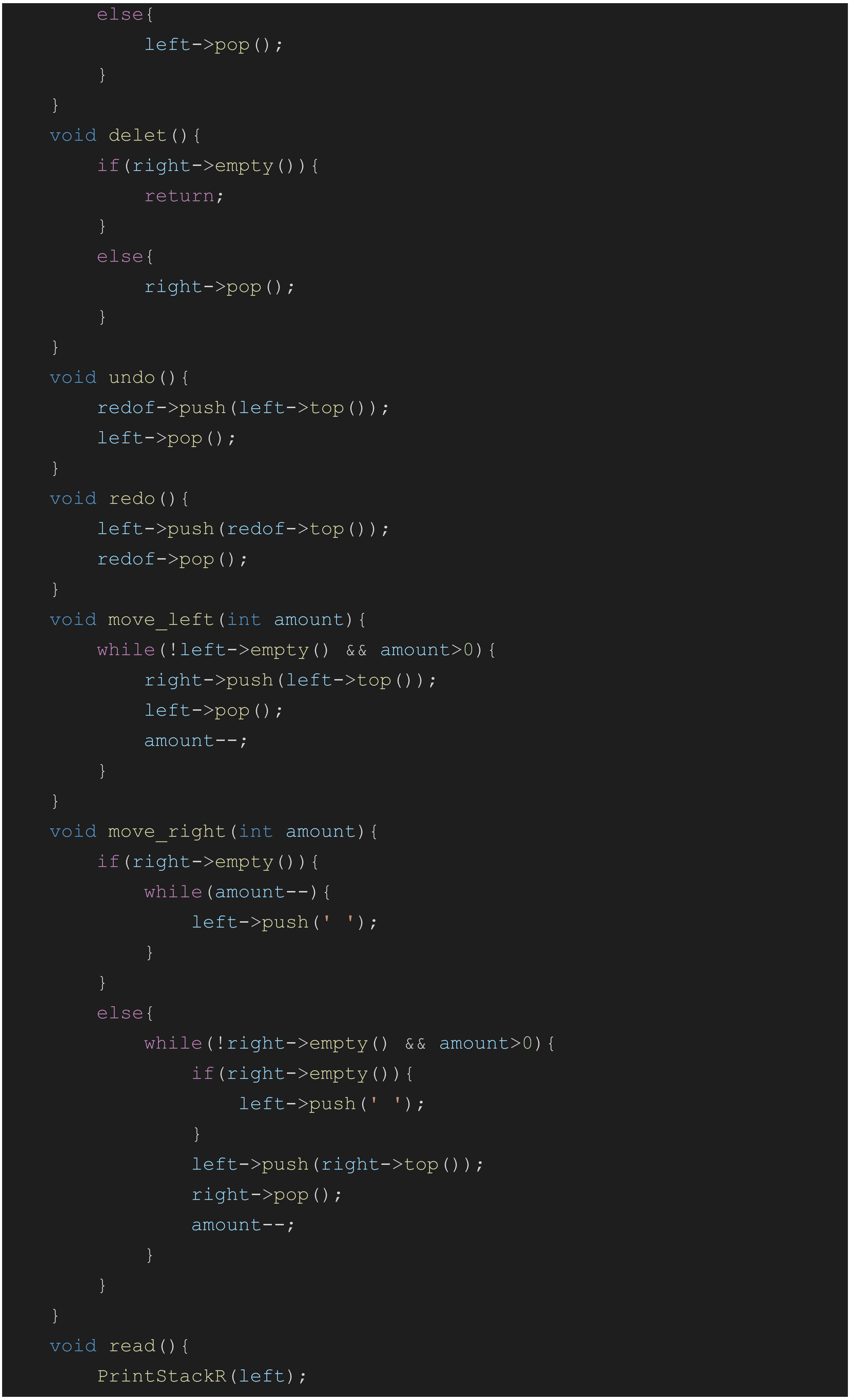
05201172020 - Khushi Punia

PROBLEM STATEMENT

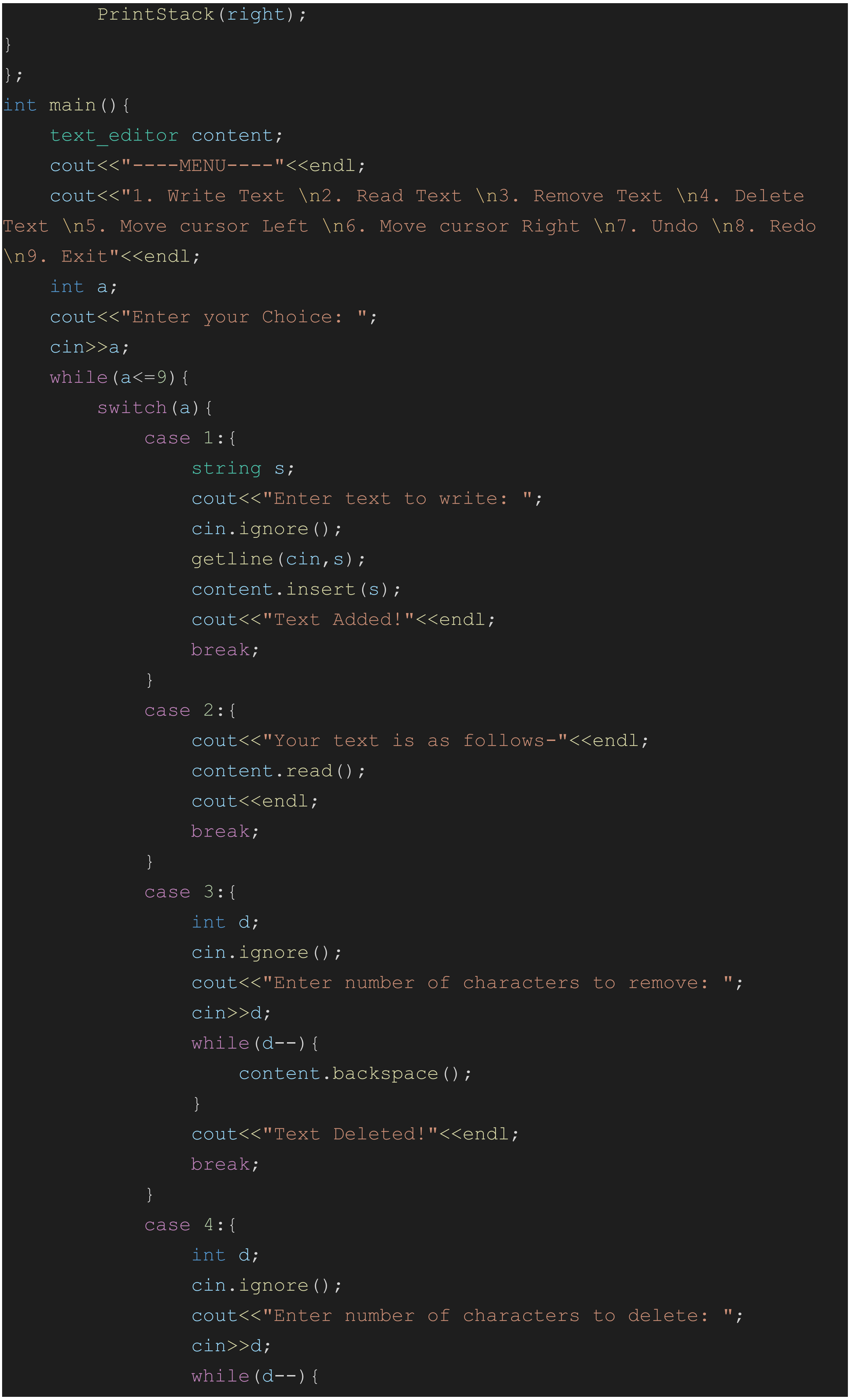
A text editor is a program that allows the user to open, view, and edit plain text files (files containing only text).
Text editors deal with manipulating text and provide features to enhance the experience. As mentioned before, the major functionalities of text editors are: inserting, deleting, and viewing text. Additional features that are practically required to even compete with other text editors are: read and write text, undo and redo functionalities, moving cursor position and deleting characters using DEL and BACKSPACE.

CODE

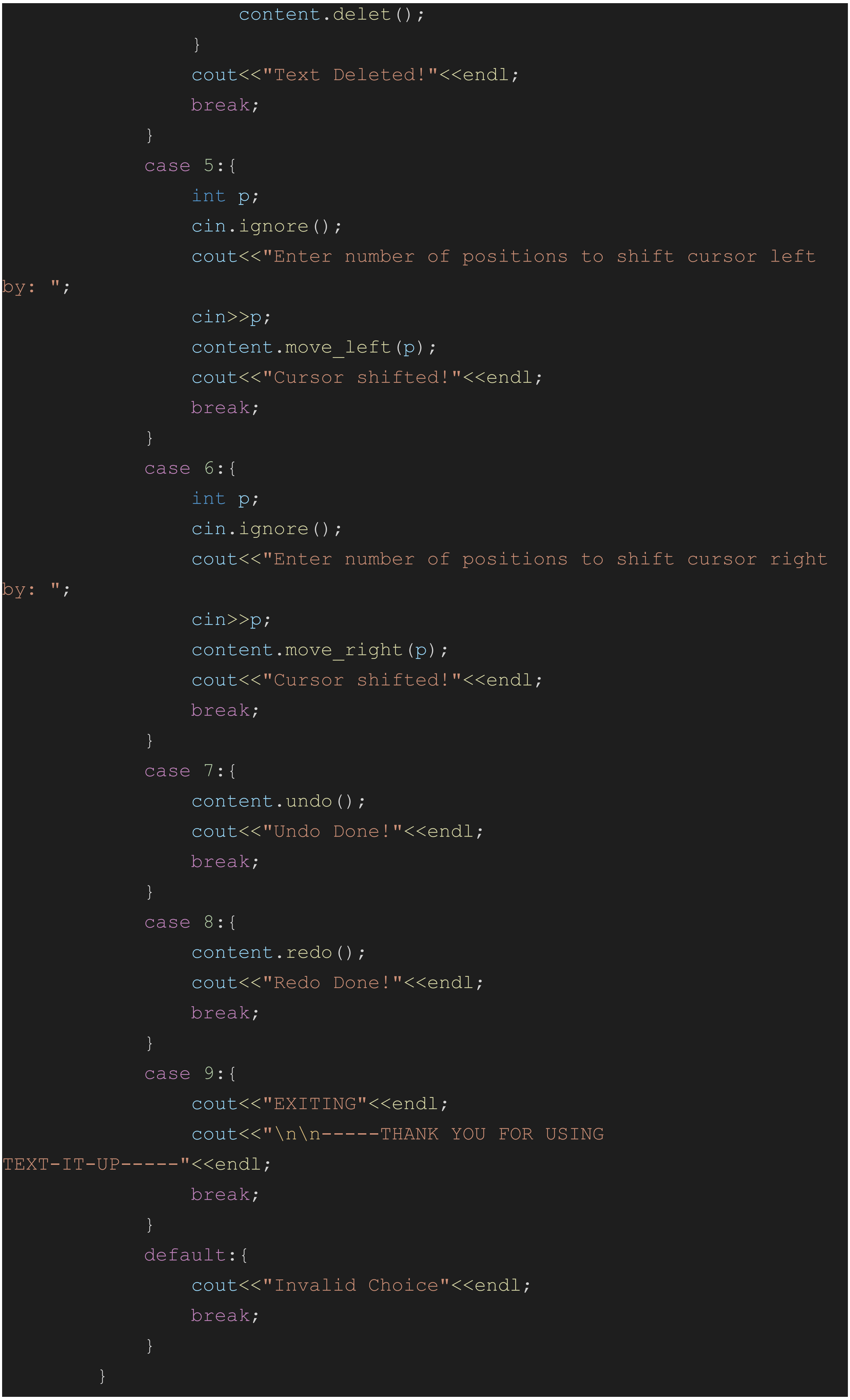
```
#include <iostream>
#include <stack>
#include<bits/stdc++.h>
using namespace std;
void PrintStackR(stack<char> *s){
    if (s->empty())
        return;
    char x = s->top();
    s->pop();
    PrintStackR(s);
    cout<<x;
    s->push(x);
}

void PrintStack(stack<char> *s){
    if (s->empty())
        return;
    char x = s->top();
    s->pop();
    cout<<x;
    PrintStack(s);
    s->push(x);
}

class text_editor{
    public:
    stack<char> *left;
    stack<char> *right;
    stack<char> *redof;
    text_editor(){
        left=new stack<char>();
        right=new stack<char>();
        redof=new stack<char>();
    }
    void insert(string input){
        for(int i=0;i<input.size();i++){
            left->push(input[i]);
        }
    }
    void backspace(){
        if(left->empty()){
            return;
        }
```



```
        else{
            left->pop();
        }
    }
    void delet(){
        if(right->empty()){
            return;
        }
        else{
            right->pop();
        }
    }
    void undo(){
        redof->push(left->top());
        left->pop();
    }
    void redo(){
        left->push(redof->top());
        redof->pop();
    }
    void move_left(int amount){
        while(!left->empty() && amount>0){
            right->push(left->top());
            left->pop();
            amount--;
        }
    }
    void move_right(int amount){
        if(right->empty()){
            while(amount--){
                left->push(' ');
            }
        }
        else{
            while(!right->empty() && amount>0){
                if(right->empty()){
                    left->push(' ');
                }
                left->push(right->top());
                right->pop();
                amount--;
            }
        }
    }
    void read(){
        PrintStackR(left);
```



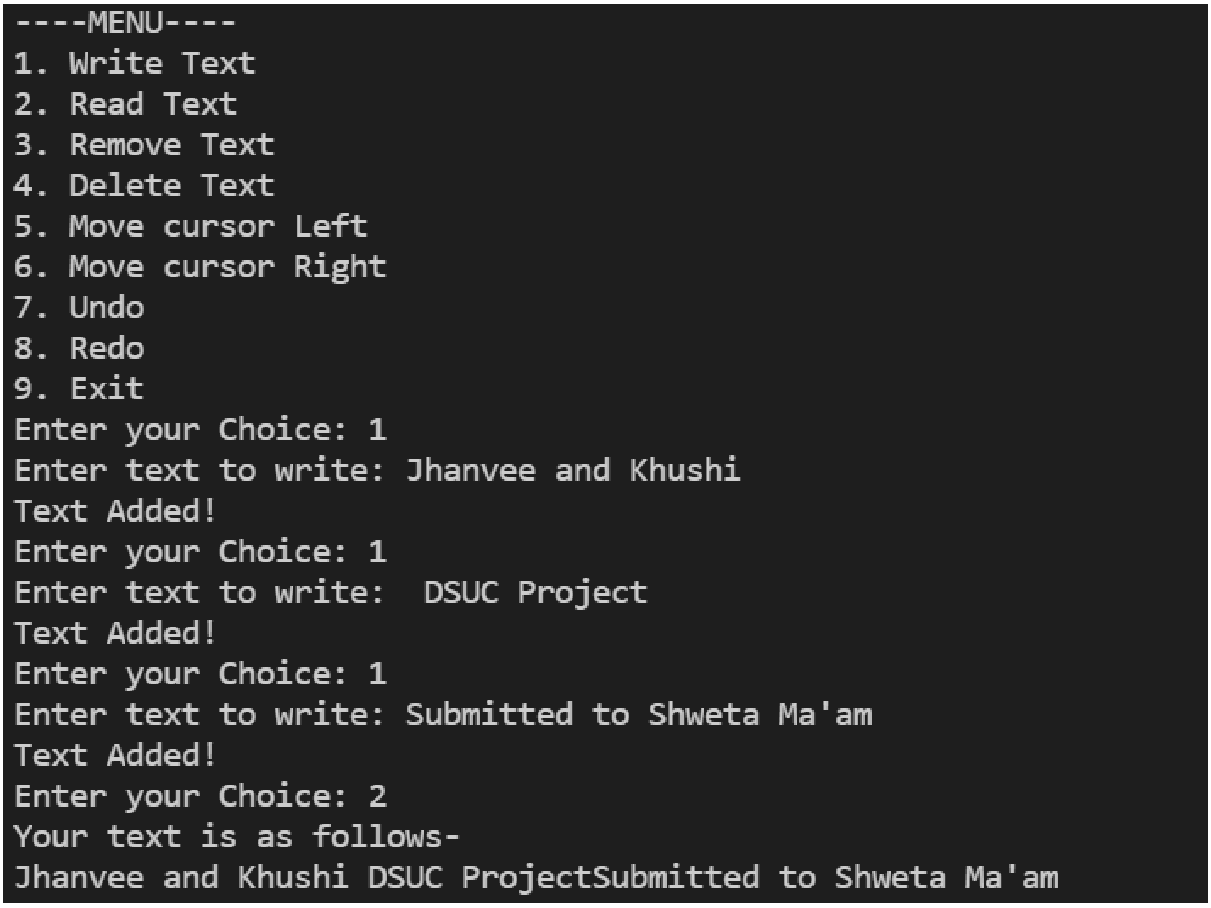
```
        PrintStack(right);
}
};
int main(){
    text_editor content;
    cout<<"----MENU----"<<endl;
    cout<<"1. Write Text \n2. Read Text \n3. Remove Text \n4. Delete Text \n5. Move cursor Left \n6. Move cursor Right \n7. Undo \n8. Redo \n9. Exit"<<endl;
    int a;
    cout<<"Enter your Choice: ";
    cin>>a;
    while(a<=9){
        switch(a){
            case 1:{
                string s;
                cout<<"Enter text to write: ";
                cin.ignore();
                getline(cin,s);
                content.insert(s);
                cout<<"Text Added!"<<endl;
                break;
            }
            case 2:{
                cout<<"Your text is as follows-"<<endl;
                content.read();
                cout<<endl;
                break;
            }
            case 3:{
                int d;
                cin.ignore();
                cout<<"Enter number of characters to remove: ";
                cin>>d;
                while(d--){
                    content.backspace();
                }
                cout<<"Text Deleted!"<<endl;
                break;
            }
            case 4:{
                int d;
                cin.ignore();
                cout<<"Enter number of characters to delete: ";
                cin>>d;
                while(d--){
```



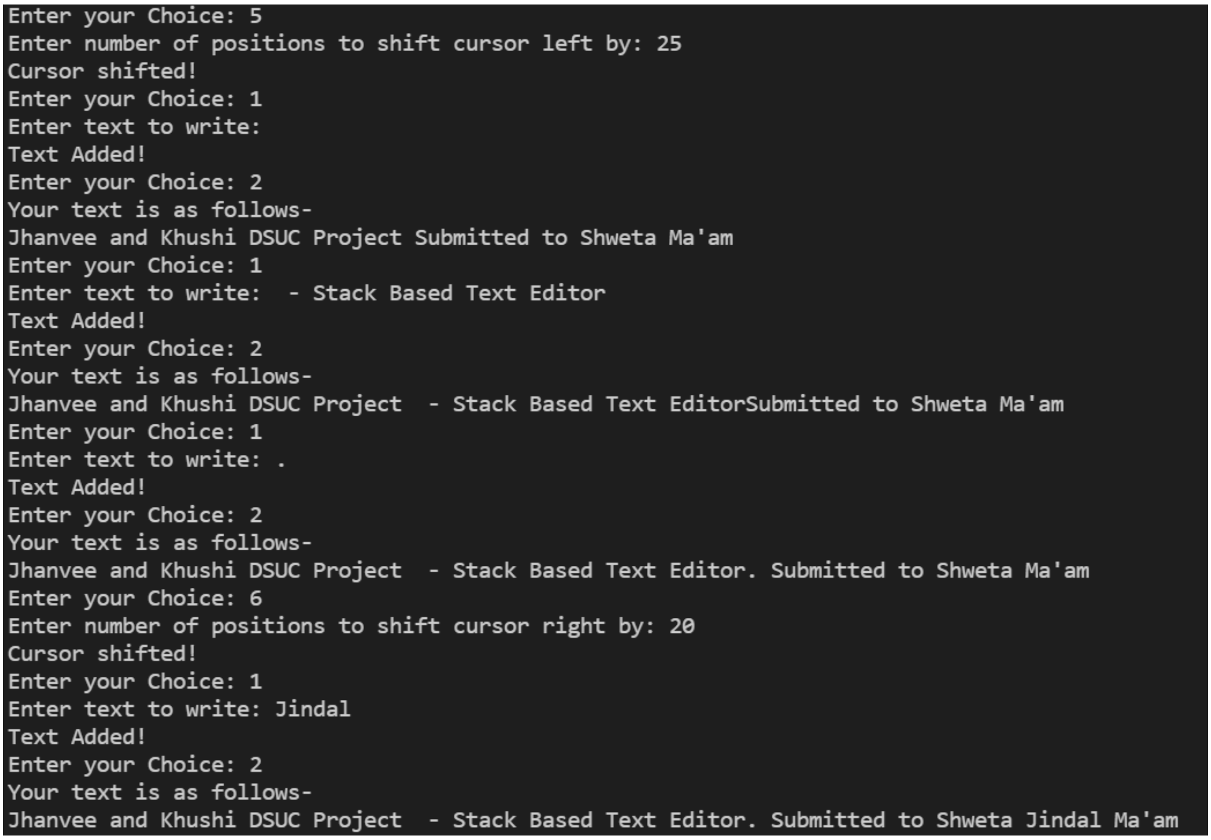
```
                    content.delet();
                }
                cout<<"Text Deleted!"<<endl;
                break;
            }
            case 5:{
                int p;
                cin.ignore();
                cout<<"Enter number of positions to shift cursor left 
by: ";
                cin>>p;
                content.move_left(p);
                cout<<"Cursor shifted!"<<endl;
                break;
            }
            case 6:{
                int p;
                cin.ignore();
                cout<<"Enter number of positions to shift cursor right 
by: ";
                cin>>p;
                content.move_right(p);
                cout<<"Cursor shifted!"<<endl;
                break;
            }
            case 7:{
                content.undo();
                cout<<"Undo Done!"<<endl;
                break;
            }
            case 8:{
                content.redo();
                cout<<"Redo Done!"<<endl;
                break;
            }
            case 9:{
                cout<<"EXITING"<<endl;
                cout<<"\n\n-----THANK YOU FOR USING 
TEXT-IT-UP-----"<<endl;
                break;
            }
            default:{
                cout<<"Invalid Choice"<<endl;
                break;
            }
        }
        if(a==9){
            break;
        }
        else{
            cout<<"Enter your Choice: ";
            cin>>a;
        }
    }
    return 0;
}
```

Github Repository : https://github.com/jhanvee-khola/TextEditor

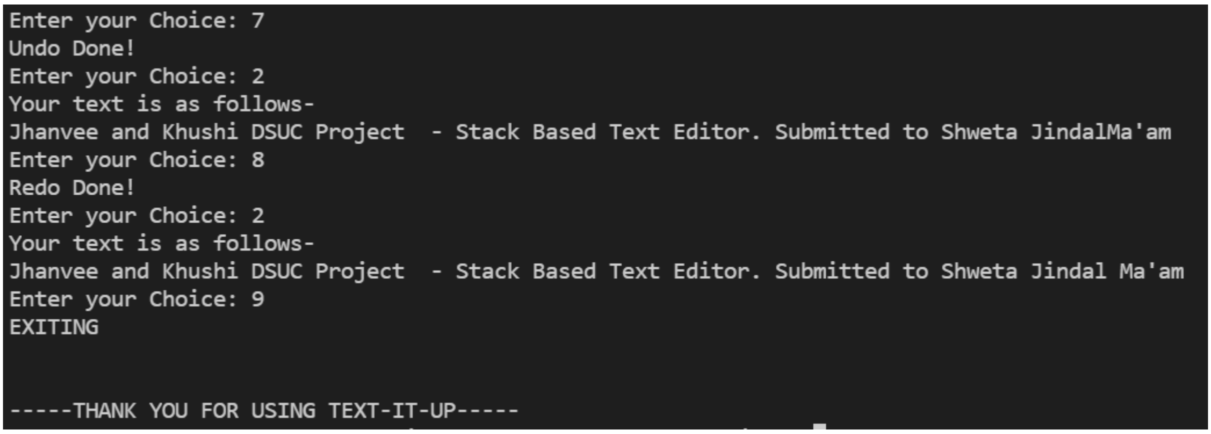
OUTPUT



```
----MENU----
1. Write Text
2. Read Text
3. Remove Text
4. Delete Text
5. Move cursor Left
6. Move cursor Right
7. Undo
8. Redo
9. Exit
Enter your Choice: 1
Enter text to write: Jhanvee and Khushi
Text Added!
Enter your Choice: 1
Enter text to write:  DSUC Project
Text Added!
Enter your Choice: 1
Enter text to write: Submitted to Shweta Ma'am
Text Added!
Enter your Choice: 2
Your text is as follows-
Jhanvee and Khushi DSUC ProjectSubmitted to Shweta Ma'am
```



```
Enter your Choice: 5
Enter number of positions to shift cursor left by: 25
Cursor shifted!
Enter your Choice: 1
Enter text to write: 
Text Added!
Enter your Choice: 2
Your text is as follows-
Jhanvee and Khushi DSUC Project Submitted to Shweta Ma'am
Enter your Choice: 1
Enter text to write:  - Stack Based Text Editor
Text Added!
Enter your Choice: 2
Your text is as follows-
Jhanvee and Khushi DSUC Project  - Stack Based Text EditorSubmitted to Shweta Ma'am
Enter your Choice: 1
Enter text to write: .
Text Added!
Enter your Choice: 2
Your text is as follows-
Jhanvee and Khushi DSUC Project  - Stack Based Text Editor. Submitted to Shweta Ma'am
Enter your Choice: 6
Enter number of positions to shift cursor right by: 20
Cursor shifted!
Enter your Choice: 1
Enter text to write: Jindal
Text Added!
Enter your Choice: 2
Your text is as follows-
Jhanvee and Khushi DSUC Project  - Stack Based Text Editor. Submitted to Shweta Jindal Ma'am
```



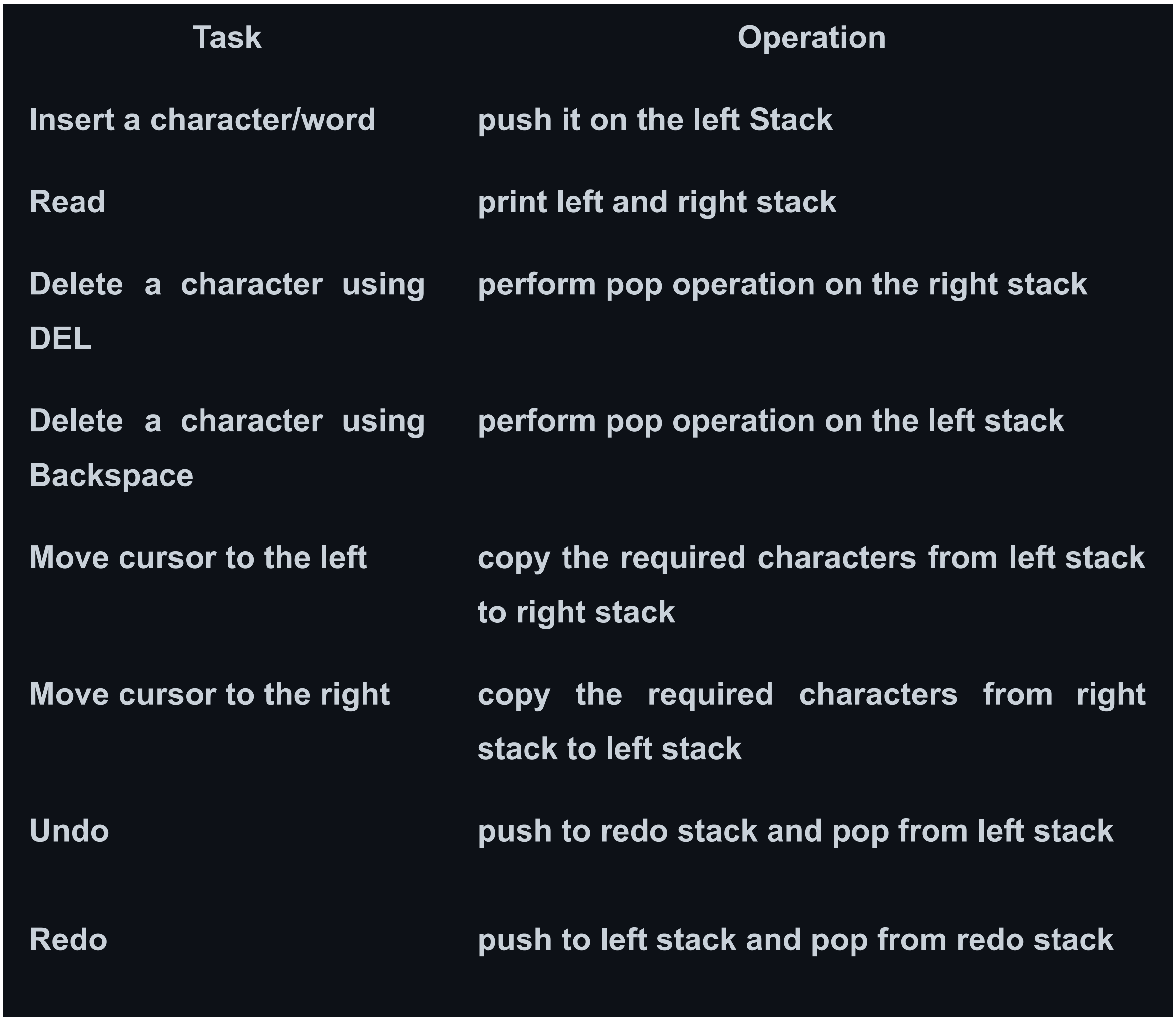
```
Enter your Choice: 7
Undo Done!
Enter your Choice: 2
Your text is as follows-
Jhanvee and Khushi DSUC Project  - Stack Based Text Editor. Submitted to Shweta JindalMa'am
Enter your Choice: 8
Redo Done!
Enter your Choice: 2
Your text is as follows-
Jhanvee and Khushi DSUC Project  - Stack Based Text Editor. Submitted to Shweta Jindal Ma'am
Enter your Choice: 9
EXITING


-----THANK YOU FOR USING TEXT-IT-UP-----
```

ALGORITHM

Two Stack Model

This approach is most likely only found in programming assignments rather than actual implementation, but it is good practice using stacks. The idea is to use two stacks to maintain the current cursor position. Left stack would contain the contents to the left of the cursor and the right stack would contain the contents to the right cursor. You would just push or pop the elements to get to where you need and either insert or delete text.

We used the cursor position as ‘divider’ and maintained two stacks to hold characters on either side. All characters to the left of the cursor are pushed on the ‘left’ stack All characters to its right are pushed on the ‘right’ stack.
Use 2 stacks to depict the text editor



Task	Operation
Insert a character/word	push it on the left Stack
Read	print left and right stack
Delete a character using DEL	perform pop operation on the right stack
Delete a character using Backspace	perform pop operation on the left stack
Move cursor to the left	copy the required characters from left stack to right stack
Move cursor to the right	copy the required characters from right stack to left stack
Undo	push to redo stack and pop from left stack
Redo	push to left stack and pop from redo stack