

Stack Text Editor

Submitted To Usman Ghani

Submitted By Khadija Javed

Roll No 1227(3M)

Project Data structure & Algorithm

Department Computer Science

**Documentation for Text Editor Code**

**Overview**

This Text Editor application is a simple C++ program that allows users to perform basic text editing operations. It supports inserting and deleting characters, navigating through the text, and undoing and redoing changes. The application uses a `vector` to store text, and `stack` structures to manage undo and redo operations efficiently.

Classes and Methods

Class: `TextEditor`

Private Members

* `vector<char> text`: Stores the text characters.
* `int cursorPosition`: Keeps track of the current cursor position in the text.
* `stack<string> undoStack`: Stores previous states of the text for undo operations.
* `stack<string> redoStack`: Stores undone states of the text for redo operations.

Private Methods

`void saveState(stack<string>& stack)`: Saves the current state of the text to the given stack.

***Public Methods***

Constructor

`TextEditor()`: Initializes the `TextEditor` object with the cursor at position 0.

Editing Methods

* `void insert(char ch)`: Inserts a character at the current cursor position, saves the current state for undo, and clears the redo stack.
* `void del()`: Deletes a character at the current cursor position if possible, saves the current state for undo, and clears the redo stack.

Undo/Redo Methods

* `void undo()`: Reverts the text to the previous state if possible and saves the current state for redo.
* `void redo()`: Reapplies the last undone state if possible and saves the current state for undo.

Cursor Navigation Methods

* `void next()`: Moves the cursor to the next position if possible.
* `void previous()`: Moves the cursor to the previous position if possible.
* `void start()`: Moves the cursor to the start of the text.
* `void end()`: Moves the cursor to the end of the text.
* `void curr\_position(int pos)`: Sets the cursor to the specified position if it is within the text bounds.

Display Method

- `void display()`: Displays the current state of the text.

Main Function

**cpp**

int main() {

TextEditor editor;

editor.insert('K');

editor.insert('H');

editor.insert('A');

editor.insert('D');

editor.insert('l');

editor.insert('J');

editor.insert('A');

editor.display();

editor.undo();

editor.display();

editor.redo();

editor.display();

return 0;

}

```

***Usage***

1. Insert Characters

- Use `insert(char ch)` to insert characters at the current cursor position.

2. Delete Characters

- Use `del()` to delete characters at the current cursor position.

3. Undo Changes

- Use `undo()` to revert the text to the previous state.

4. Redo Changes

- Use `redo()` to reapply the last undone state.

5. Navigate Through Text

1. Use `next()` to move the cursor forward.
2. Use `previous()` to move the cursor backward.
3. Use `start()` to move the cursor to the beginning of the text.
4. Use `end()` to move the cursor to the end of the text.
5. Use `curr\_position(int pos)` to set the cursor to a specific position.

6. Display Text

- Use `display()` to print the current state of the text.

Example

In the `main` function, the following operations are performed:

- Insert the characters ‘K’,'H', 'A', 'D', 'l',’J’, 'A' into the text.

- Display the text, showing "Khadija".

- Undo the last insertion, resulting in "khadij".

- Redo the undone insertion, restoring the text to "Khadija".