

Text Editor



→ display  
→ insert  
↳ (n)

display → (5, 10)

array → 0 → 1, 1 → 2

→ Undo  
↳ stack

→ Redo  
↳ stack

7 → New Ground

delete  
↳ line → 7

↳ (7 - 11)

↳ copy  
(n, m)

Paste  
↳ (n, m)  
↳ (multiple characters)

1 My name is Kartikya  
2 \_\_\_\_\_  
3 \_\_\_\_\_  
4 \_\_\_\_\_  
5 \_\_\_\_\_  
6 \_\_\_\_\_  
7 \_\_\_\_\_  
8 \_\_\_\_\_  
9 \_\_\_\_\_  
10 \_\_\_\_\_  
11 \_\_\_\_\_  
12 \_\_\_\_\_  
13 \_\_\_\_\_  
14 \_\_\_\_\_  
15 \_\_\_\_\_  
16 \_\_\_\_\_

## ↳ Requirements

- 1) Display the text
- 2) Display the text from given lines →
- 3) Insert the text
- 4) Delete the text
- 5) Delete the text from the given lines (n, m)
- 6) Copy the text →
- 7) Paste the text
- 8) Undo your action
- 9) Redo your action
- 10) Exit



→ H.W. Insertion in multiple lines

(for(int i = startline; i <= ending; i++)  
insert

→ H.W.

Other than these operations

↳ Invalid operation

## ↳ Notepad class

↳ delete, insert, copy, paste, etc.

↳ 100.

↳ is

↳ implementation lines → 100 (capacity)

Action class

↳ string arr [ ] → size = capacity      actionId → int.

↳ Stack → undo → Stack < ~~String~~ > < Action >

↳ Stack → redo

↳ queue → copy

↳ String → dateTime → Local Datetime  
↳ String → line no. → int  
↳ String → text → string  
↳ String → is Addition → boolean

## Action class

↳ action Id → int

↳ date Time → LocalDateTime

↳ line no. → int.

↳ test → string

↳ is Addition → boolean

} → private  
private is activated

↳ getter/setter →

Display the text from given lines

↳ starting, ending

1)  $\left( \begin{array}{l} \text{starting} > \text{ending} \quad | \quad \text{starting} < 0 \quad | \quad \text{ending} > \text{capacity} \end{array} \right)$

Starting line  $\rightarrow$  `unordered(StartingLine - 1)`

Insert line  
↳ (line no, text)

( if (line no. > 100)

notepad[line no - 1] = text;

undoAction .



- (a) Check for valid line no -
- (b) Insert line
- (c) Add the action to undo stack

Delete → (line no.)

↳ line no. > capacity → error

↳ \_\_\_\_\_

↳ `notepad[line no - 1] = ""`

↳ redo stack

1) check for valid line no -

2) `notepad[line no - 1]` not empty.

3) `notepad[line no - 1] = ""` and push action in Redo stack.



Delete multiple lines → (start line, ending line)

↳ check for valid line no.

↳ for (i = starting line; i <= ending line; i++)

}

delete (i);

}

→ Copy (startLine, endLine)

↳ Invalid.

↳ startLine > endLine

|| endLine > capacity

(a) Invalid checks

(b) Copy the data in copyText.

(c) add to the queue.

String copyText = "";

for (int i = startLine - 1; i < endLine; i++)

copyText = copyText + notepad[i];

if (!copyText.isEmpty())

clipboard.add(copyText);

→ Paste the text (int line no.)

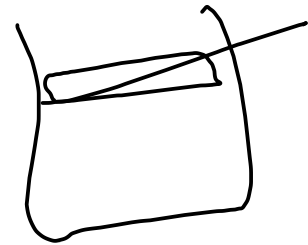
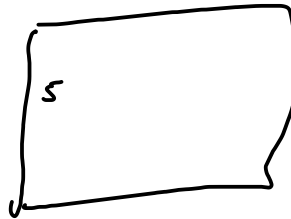
↳ Clip Board is empty → return

↳ string text = clipBoard.peek();

↳ insert(line no, text);

## Undo the action

- (a) check if stack is empty or not.
  - (b) Take out the later + action and pop from the stack also  
↳ line
  - (c) Delete (line no.)
- action a = → line no. → delete fr.



## Redo

- ↳ (1) stack is empty or not
- ↳ (2) Pick the action and pop from the stack.
- ↳ (3) Insert (line no, text)