Text Editor using Two Stack Model

Data Structure used : Stack [Two Stack Model]

In this data stucture, there are two stacks used hence the name two stack model. The idea behind this data structure is to mimic editing lines rather than a huge document.

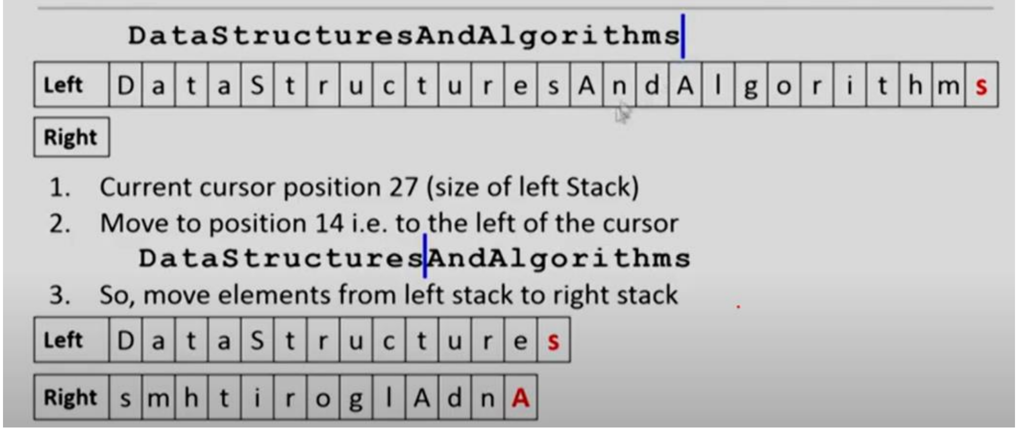
Working of Text Editor:

The idea is to use two stacks to maintain the current cursor position. We used the cursor position as 'divider' and maintained two stacks to hold characters on either side. All characters to the left of cursor are pushed on the 'left' stack and All characters to its right, are pushed on the 'right' stack. You would just push or pop the elements to get to where you need and either insert or delete text.

Functionality of different Functions:

Task	**Operation**
Insert a character / word.	push it on the left 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
To replace	Use a combination of move cursor with DEL and insert operations

Move Cursor (Left) :



DataStructuresAndAlgorithms

Left	D	a	t	a	S	t	r	u	c	t	u	r	e	s	A	n	d	A	l	g	o	r	i	t	h	m	s
Right																											

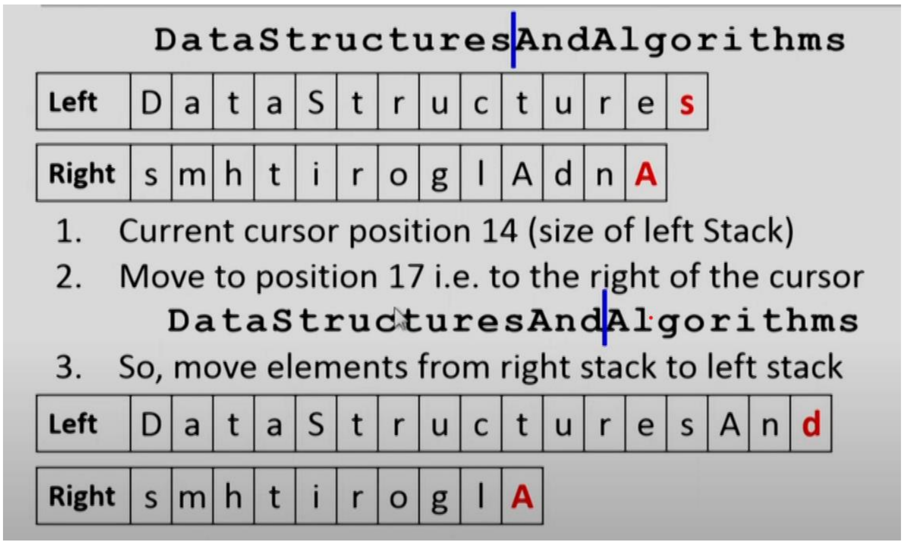
1. Current cursor position 27 (size of left Stack)
2. Move to position 14 i.e. to the left of the cursor

   DataStructuresAndAlgorithms

3. So, move elements from left stack to right stack

Left	D	a	t	a	S	t	r	u	c	t	u	r	e	s
Right	s	m	h	t	i	r	o	g	l	A	d	n	A	

Move Cursor (Right) :



DataStructuresAndAlgorithms

Left	D	a	t	a	S	t	r	u	c	t	u	r	e	s
Right	s	m	h	t	i	r	o	g	l	A	d	n	A	

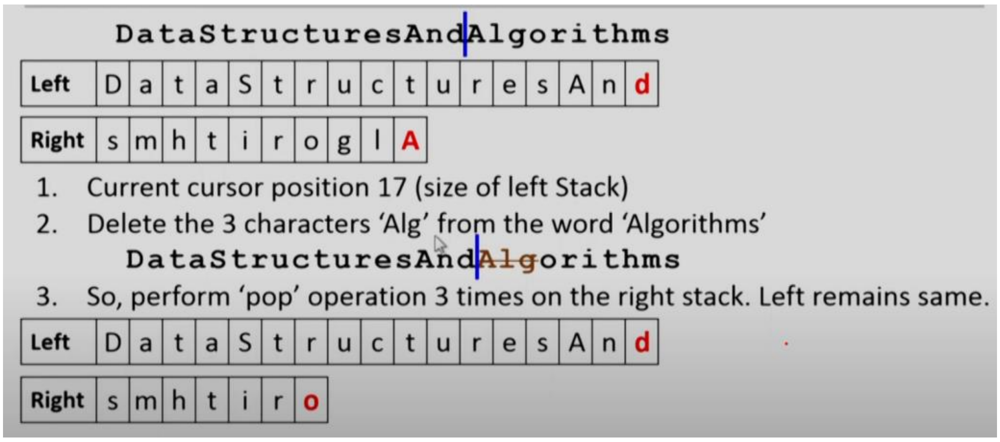
1. Current cursor position 14 (size of left Stack)
2. Move to position 17 i.e. to the right of the cursor

   DataStructuresAndAlgorithms

3. So, move elements from right stack to left stack

Left	D	a	t	a	S	t	r	u	c	t	u	r	e	s	A	n	d
Right	s	m	h	t	i	r	o	g	l	A							

Delete Character:



DataStructuresAndAlgorithms

Left	D	a	t	a	S	t	r	u	c	t	u	r	e	s	A	n	d
Right	s	m	h	t	i	r	o	g	l	A							

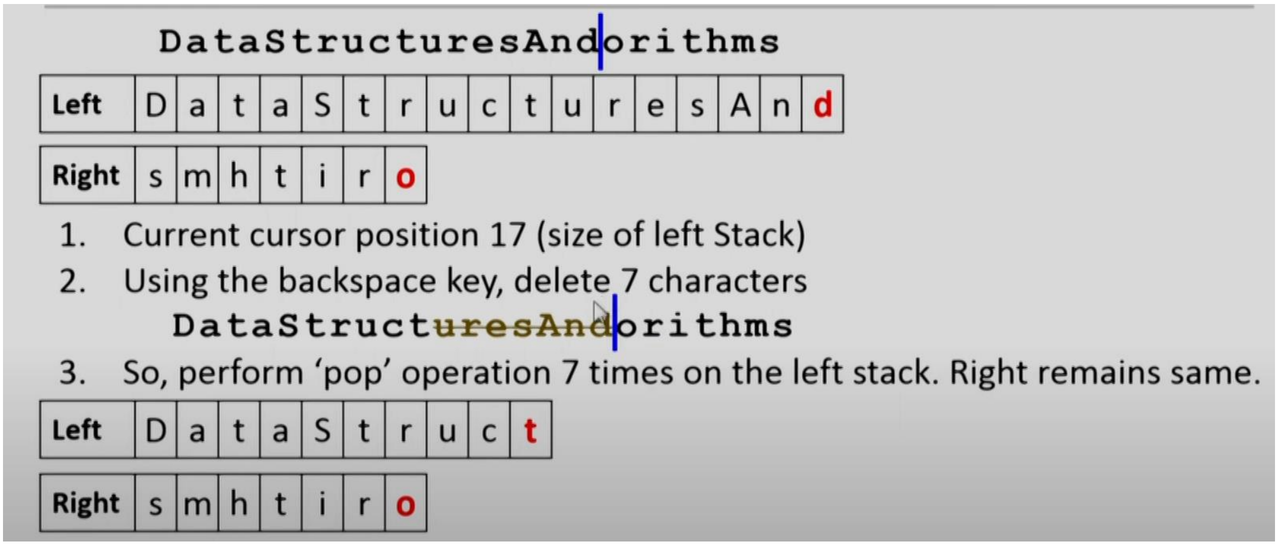
1. Current cursor position 17 (size of left Stack)
2. Delete the 3 characters 'Alg' from the word 'Algorithms'

   DataStructuresAnd~~Alg~~orithms

3. So, perform 'pop' operation 3 times on the right stack. Left remains same.

Left	D	a	t	a	S	t	r	u	c	t	u	r	e	s	A	n	d
Right	s	m	h	t	i	r	o										

Backspace Character:



DataStructuresAndorithms

Left	D	a	t	a	S	t	r	u	c	t	u	r	e	s	A	n	**d**

Right	s	m	h	t	i	r	**o**

1. Current cursor position 17 (size of left Stack)
2. Using the backspace key, delete 7 characters

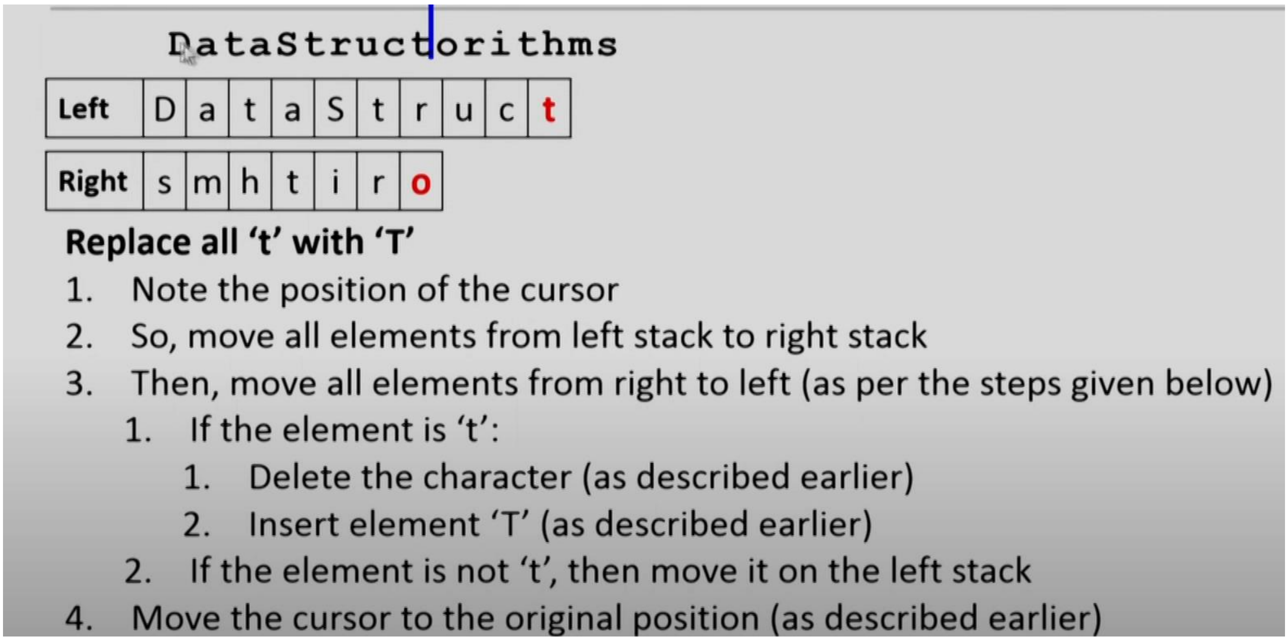
   DataStruct~~uresAnd~~orithms

3. So, perform 'pop' operation 7 times on the left stack. Right remains same.

Left	D	a	t	a	S	t	r	u	c	**t**

Right	s	m	h	t	i	r	**o**

Find And Replace Character:



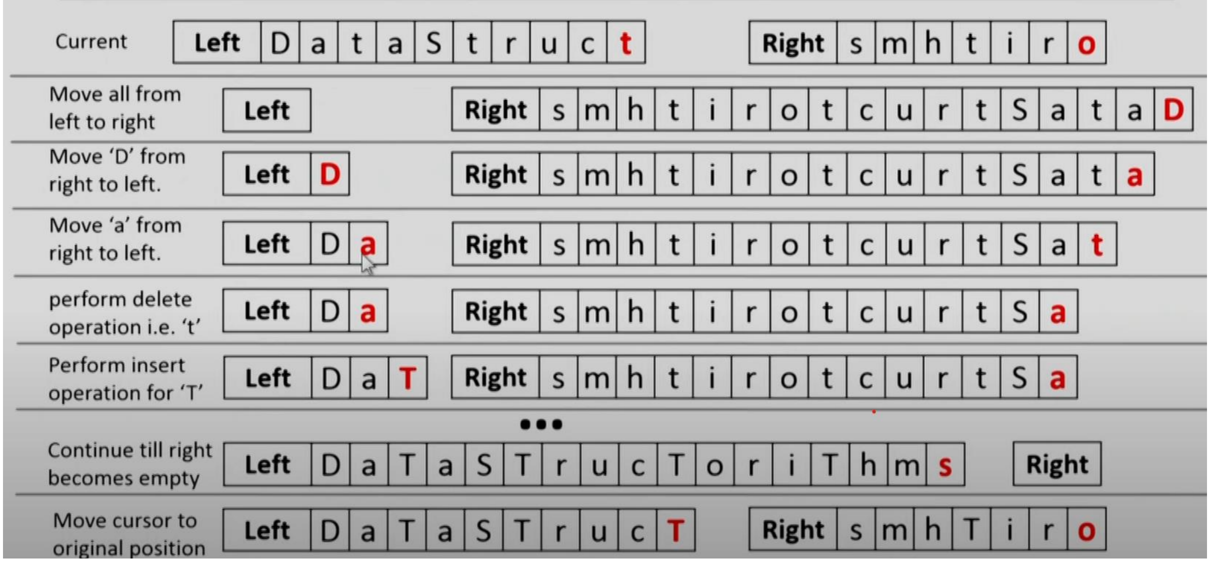
DataStructorithms

Left	D	a	t	a	S	t	r	u	c	**t**

Right	s	m	h	t	i	r	**o**

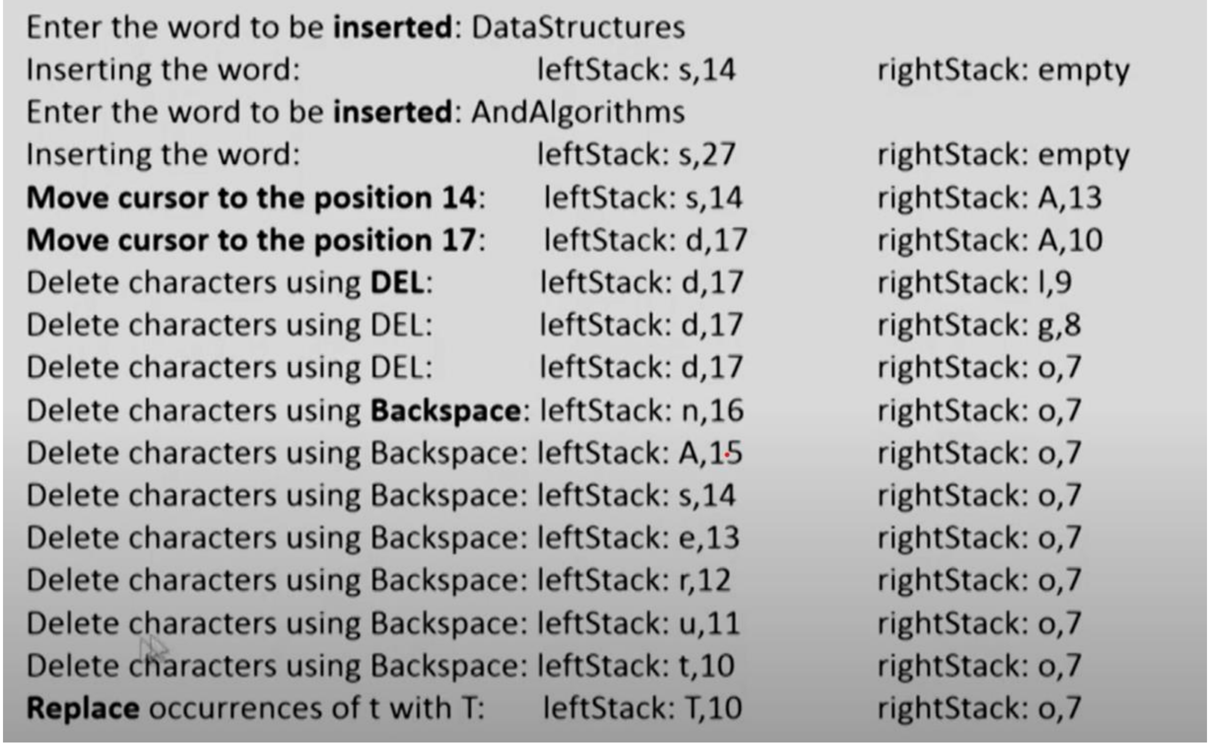
Replace all 't' with 'T'

1. Note the position of the cursor
2. So, move all elements from left stack to right stack
3. Then, move all elements from right to left (as per the steps given below)
   1. If the element is 't':
      1. Delete the character (as described earlier)
      2. Insert element 'T' (as described earlier)
   2. If the element is not 't', then move it on the left stack
4. Move the cursor to the original position (as described earlier)



Current	Left D a t a S t r u c t	Right s m h t i r o
Move all from left to right	Left	Right s m h t i r o t c u r t S a t a D
Move 'D' from right to left.	Left D	Right s m h t i r o t c u r t S a t a
Move 'a' from right to left.	Left D a	Right s m h t i r o t c u r t S a t
perform delete operation i.e. 't'	Left D a	Right s m h t i r o t c u r t S a
Perform insert operation for 'T'	Left D a T	Right s m h t i r o t c u r t S a
•••		
Continue till right becomes empty	Left D a T a S T r u c T o r i T h m s	Right
Move cursor to original position	Left D a T a S T r u c T	Right s m h T i r o

Example:



```
Enter the word to be inserted: DataStructures
Inserting the word:                   leftStack: s,14          rightStack: empty
Enter the word to be inserted: AndAlgorithms
Inserting the word:                   leftStack: s,27          rightStack: empty
Move cursor to the position 14:       leftStack: s,14          rightStack: A,13
Move cursor to the position 17:       leftStack: d,17          rightStack: A,10
Delete characters using DEL:          leftStack: d,17          rightStack: l,9
Delete characters using DEL:          leftStack: d,17          rightStack: g,8
Delete characters using DEL:          leftStack: d,17          rightStack: o,7
Delete characters using Backspace:    leftStack: n,16          rightStack: o,7
Delete characters using Backspace:    leftStack: A,15          rightStack: o,7
Delete characters using Backspace:    leftStack: s,14          rightStack: o,7
Delete characters using Backspace:    leftStack: e,13          rightStack: o,7
Delete characters using Backspace:    leftStack: r,12          rightStack: o,7
Delete characters using Backspace:    leftStack: u,11          rightStack: o,7
Delete characters using Backspace:    leftStack: t,10          rightStack: o,7
Replace occurrences of t with T:      leftStack: T,10          rightStack: o,7
```