

Formal Methods (形式化方法)

Lecture 10. An Application of Sequence (序列的应用)

智能与计算学部 章衡

2021年上学期



1 A Simple Text Editor

2 Extended Text Editor



Outline

1 A Simple Text Editor

2 Extended Text Editor



A simple text editor: Types and descriptions of states

[Char, Report]

Report ::= “Okay” | “At top of document” | “At bottom of document”

Doc1 _____

left, right : seq Char

Δ Doc1 _____

Doc1

Doc1'

\exists Doc1 _____

Δ Doc1

left = left'

right = right'



A simple text editor: Descriptions of operations (1)

InitDoc1 _____

Doc1'

left' = $\langle \rangle$

right' = $\langle \rangle$

Success _____

rep! : Report

rep! = “Okay”



A simple text editor: Descriptions of operations (2)

DeleteLeftDoc1

ΔDoc1

$\text{left} \neq \langle \rangle$

$\text{left}' = \text{front left}$

$\text{right}' = \text{right}$

ErrorAtTop

$\exists\text{Doc1}$

$\text{rep!} : \text{Report}$

$\text{left} = \langle \rangle$

$\text{rep!} = \text{"At top of document"}$

$\text{DoDeleteLeftDoc1} \triangleq (\text{DeleteLeftDoc1} \wedge \text{Success}) \vee \text{ErrorAtTop}$



A simple text editor: Descriptions of operations (3)

 MoveLeftDoc1 ΔDoc1 $\text{left} \neq \langle \rangle$ $\text{left}' = \text{front left}$ $\text{right}' = \langle \text{last left} \rangle \frown \text{right}$
$$\text{DoMoveLeftDoc1} \triangleq (\text{MoveLeftDoc1} \wedge \text{Success}) \vee \text{ErrorAtTop}$$


A simple text editor: Descriptions of operations (4)

DeleteRightDoc1 _____

ΔDoc1

$\text{right} \neq \langle \rangle$

$\text{right}' = \text{tail right}$

$\text{left}' = \text{left}$

ErrorAtBottom _____

ΞDoc1

$\text{rep!} : \text{Report}$

$\text{right} = \langle \rangle$

$\text{rep!} = \text{"At bottom of document"}$

$\text{DoDeleteRightDoc1} \triangleq (\text{DeleteRightDoc1} \wedge \text{Success}) \vee \text{ErrorAtTBottom}$



A simple text editor: Descriptions of operations (5)

MoveRightDoc1 _____

ΔDoc1

$\text{right} \neq \langle \rangle$

$\text{left}' = \text{left} \cap \langle \text{head right} \rangle$

$\text{right}' = \text{tail right}$

$\text{DoMoveRightDoc1} \triangleq (\text{MoveRightDoc1} \wedge \text{Success}) \vee \text{ErrorAtBottom}$



A simple text editor: Descriptions of operations (6)

InsertLeftDoc1 _____

ΔDoc1

$\text{ch?} : \text{Char}$

$\text{left}' = \text{left} \frown \langle \text{ch?} \rangle$

$\text{right}' = \text{right}$

InsertRightDoc1 _____

ΔDoc1

$\text{ch?} : \text{Char}$

$\text{right}' = \langle \text{ch?} \rangle \frown \text{right}$

$\text{left}' = \text{left}$

$\text{DoInsertLeftDoc1} \triangleq \text{InsertLeftDoc1} \wedge \text{Success}$

$\text{DoInsertRightDoc1} \triangleq \text{InsertRightDoc1} \wedge \text{Success}$



Outline

1 A Simple Text Editor

2 Extended Text Editor



Extended text editor: Unbounded display

Line == seq(Char \ {\n})

	1	2	3	4	5	6	7	8	9	10	11
1											
2											
3											
4											
5											
6											

Text editor with unbounded display

UnboundedDisplay

doclines : seq₁ Line

ln, col : ℕ₁

ln ≤ #doclines

col ≤ #(doclines ln) + 1



Extended text editor: Operator flatten

$$\text{flatten} : \text{seq}_1 \text{ Line} \rightarrow \text{seq Char}$$

$$\forall \text{ln} : \text{Line}; \text{lns} : \text{seq}_1 \text{ Line} \bullet$$

$$\text{flatten} \langle \text{ln} \rangle = \text{ln} \wedge$$

$$\text{flatten}(\langle \text{ln} \rangle \frown \text{lns}) = (\text{ln} \frown \langle \backslash n \rangle) \frown (\text{flatten lns})$$

Example (flatten)

Let $\text{lns} == \langle \langle \text{a}, \text{b} \rangle, \langle \text{c}, \text{d}, \text{e} \rangle, \langle \text{f} \rangle \rangle$. Then we have

$$\text{flatten lns} = \langle \text{a}, \text{b}, \backslash n, \text{c}, \text{d}, \text{e}, \backslash n, \text{f} \rangle.$$

- flatten is a bijection from $\text{seq}_1 \text{ Line}$ to seq Char ,
- Thus flatten^\sim is also a total function from seq Char to $\text{seq}_1 \text{ Line}$.

Extended text editor: Descriptions of states

Doc2

Doc1

UnboundedDisplay

 $\text{left} \wedge \text{right} = \text{flatten doclines}$ $\text{ln} = \#(\text{left} \triangleright \{\backslash n\}) + 1$ $\text{col} = \# \text{left} - \#(\text{flatten } (1..(\text{ln} - 1) \upharpoonright \text{doclines}))$ 

Extended text editor: Examples

Example

Let $\text{left} == \langle a, b, \backslash n, c, d, e, \backslash n, f \rangle$, $\text{right} == \langle g, h, \backslash n, i, j \rangle$. Then

$$\text{flatten}^{\sim} \text{left} = \langle \langle a, b \rangle, \langle c, d, e \rangle, \langle f \rangle \rangle$$

Since $\text{left} \frown \text{right} = \text{flatten doclines}$, we thus have

$$\text{doclines} = \text{flatten}^{\sim} (\text{left} \frown \text{right}) = \langle \langle a, b \rangle, \langle c, d, e \rangle, \langle f, g, h \rangle, \langle i, j \rangle \rangle$$



Extended text editor: Descriptions of operations

$$\text{DoDeleteLeftDoc2} \triangleq \text{DoDeleteLeftDoc1} \wedge \Delta\text{Doc2}$$

$$\text{DoMoveLeftDoc2} \triangleq \text{DoMoveLeftDoc1} \wedge \Delta\text{Doc2}$$

$$\text{DoInsertLeftDoc2} \triangleq \text{DoInsertLeftDoc1} \wedge \Delta\text{Doc2}$$

$$\text{DoDeleteRightDoc2} \triangleq \text{DoDeleteRightDoc1} \wedge \Delta\text{Doc2}$$

$$\text{DoMoveRightDoc2} \triangleq \text{DoMoveRightDoc1} \wedge \Delta\text{Doc2}$$

$$\text{DoInsertRightDoc2} \triangleq \text{DoInsertRightDoc1} \wedge \Delta\text{Doc2}$$



Assignment

We change the definition of Report as follows:

$$\text{Report} ::= \text{“Okay”} \mid \text{“Nonexist”} \mid \text{“At top of document”} \mid \text{“At bottom of document”}$$

Please design a schema to describe the operation of search, defined as follows:

Given a character $x?$ as input, let the output be “Nonexist” if $x?$ does not appear in the text, and “Okay” otherwise. In addition, let the cursor stays at the first occurrence of $x?$ (i.e., the first character in “right” is $x?$, and $x?$ does not appear in “left”) if the output is “Okay”.

