

Operation: begin

TOP OF STACK

FRONT OF DEQUE

0

op

op

op

cl

cl

\$

Operation: shift op to stack, goto state 1

TOP OF STACK **FRONT OF DEQUE**

0

1

op

op

op

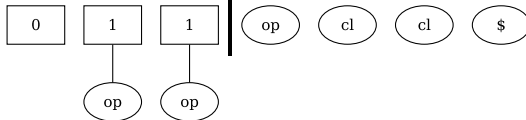
cl

cl

\$

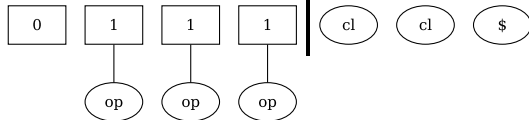
Operation: shift op to stack, goto state 1

TOP OF STACK **FRONT OF DEQUE**



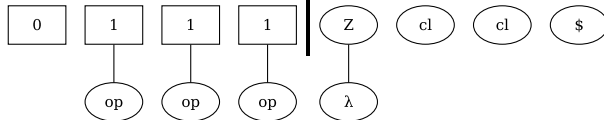
Operation: shift op to stack, goto state 1

TOP OF STACK **FRONT OF DEQUE**



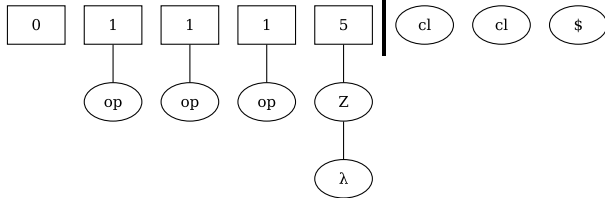
Operation: reduce by rule 5 $Z \rightarrow \lambda$

TOP OF STACK | **FRONT OF DEQUE**



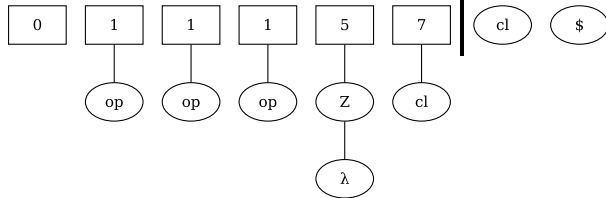
Operation: shift Z to stack, goto state 5

TOP OF STACK | **FRONT OF DEQUE**



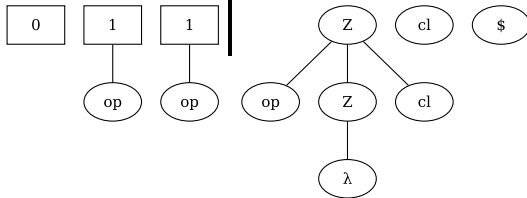
Operation: shift cl to stack, goto state 7

TOP OF STACK | **FRONT OF DEQUE**



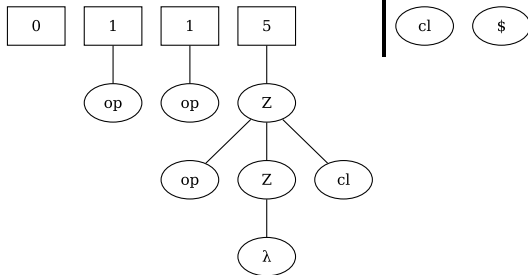
Operation: reduce by rule 4 $Z \rightarrow op\ Z\ cl$

TOP OF STACK | **FRONT OF DEQUE**



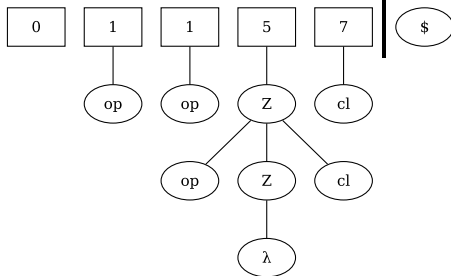
Operation: shift Z to stack, goto state 5

TOP OF STACK **FRONT OF DEQUE**



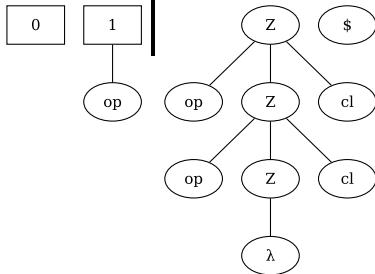
Operation: shift cl to stack, goto state 7

TOP OF STACK | **FRONT OF DEQUE**



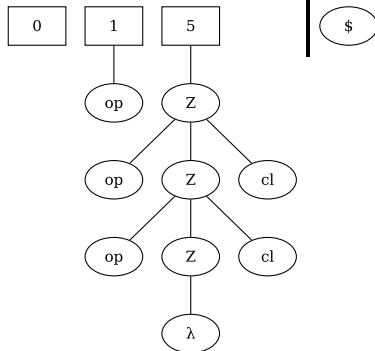
Operation: reduce by rule 4 $Z \rightarrow op\ Z\ cl$

TOP OF STACK **FRONT OF DEQUE**



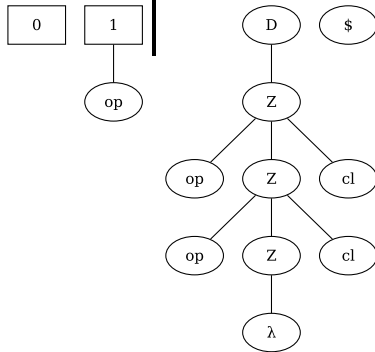
Operation: shift Z to stack, goto state 5

TOP OF STACK | **FRONT OF DEQUE**



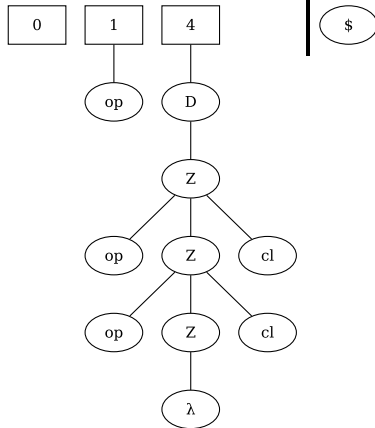
Operation: reduce by rule 3 $D \rightarrow Z$

TOP OF STACK FRONT OF DEQUE



Operation: shift D to stack, goto state 4

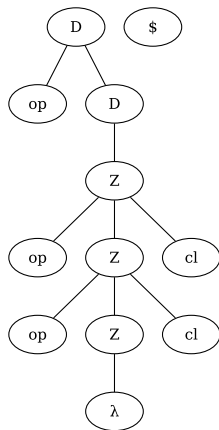
TOP OF STACK | **FRONT OF DEQUE**



Operation: reduce by rule 2 $D \rightarrow op D$

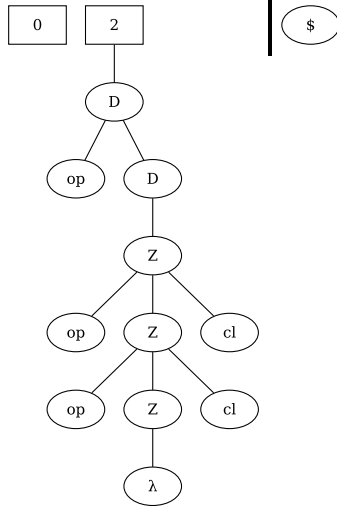
TOP OF STACK FRONT OF DEQUE

0



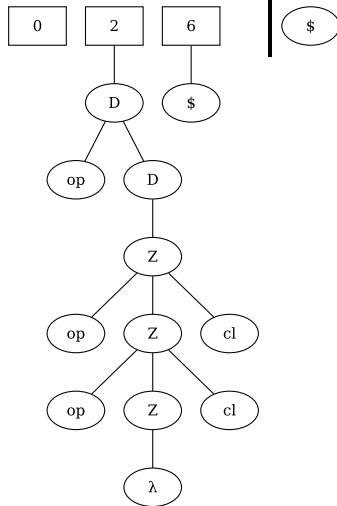
Operation: shift D to stack, goto state 2

TOP OF STACK | **FRONT OF DEQUE**



Operation: shift \$ to stack, goto state 6

TOP OF STACK, FRONT OF DEQUE



Operation: reduce by rule 1 $S \rightarrow D \$$

TOP OF STACK FRONT OF DEQUE

0

