

# Line Transformations, Part 1: Operations and Operation Sequences

Previous articles show how color complementation of lines (rows / columns) of a pattern can be used to produce interesting patterns [1,2].

Complementation is only one of many possible operations that can be used to transform lines. Shifts (cyclic permutations) and reversal are two that are particularly promising. Combinations of operations, such as both complementation and reversal, also can produce interesting results.

Like complementation, characters can be used to represent operations and sequences of characters can be used to represent operations on successive lines.

## Basic Operations

Picking characters to represent operations is somewhat arbitrary. What follows is designed to provide mnemonic help.

The basic transformations are

.	no operation
~	complementation
	reversal
$n$	shift by $n$

By default, shifts are incremental, each one adding to the previous one. The initial shift prior to transformation is 0.

Since shifts greater than 9 may be needed, the lowercase letters are used: a for 10, b for 11, and on to z for 35.

The initial direction for shifts is to the right. The direction can be changed to left by the character < and back to the right by >.

The notion of shift direction and use of < and > introduce two new concepts: a global *state* that affects the interpretation of the shift direction and characters that change the state but do not operate of the pattern.

## Operation Sequences

Sequences of characters represent sequences of operations on successive lines of a pattern (except for state-changing characters).

For example, the operation sequence

~.|

complements the first line, leaves the second unchanged, and reverses the third line.

If an operation sequence is shorter than the number of lines, it is repeated as necessary to transform all lines.

## Combined Operations

Combined operations on a line are enclosed in braces. For example, the operation sequence

{~|}.

leaves the first line unchanged, complements and reverses the second line, and leaves the third line unchanged.

The operations in a combined operation are performed in the order they appear from left to right.

## Permutations

Reversal and shifts are permutations (rearrangements) that can be applied to lines of any length.

Some permutations depend on the length of a line. These can be described by sequences that show the rearrangement of the elements of lines. For example, for a line of length 8, the permutation

87654321

is reversal, while

78123456

is a right shift of 2.

A permutation is repeated to permute a longer line, provided that the length of the permutation evenly divides the length of the line. Otherwise, it is an invalid operation. For example, the permutation 21 swaps successive pairs of elements in a line, provided the length of the line is even. For a line of length 8, this permutation is equivalent to

21426587

Permutations can eliminate and repeat ele-

ments of a line, provided the length of the result is the length of the line. For example,

11335577

duplicates odd-numbered elements of a line while eliminating even-numbered elements.

Permutations in operation sequences are enclosed in parentheses, as in

~.(21)~

which complements the first line, leaves the second line unchanged, swaps successive pairs of elements of the third line (provided the length of the line is even), and complements the fourth line.

## Defined Operations

Uppercase letters are used for defining combined operations and permutations. For example,

A={~|}

defines A to be the combined operation of complementation and reversal of a line and

P=(21)

defines P to be the permutation that swaps successive elements.

Combined operations can be used in other combined operations. For example,

B={AP}

defines an operation that complements, reverses, and swaps successive pairs of elements of a line.

## Shifts

By default, shifts are incremental and add to the previous shift. Sometimes absolute shifts, which are independent of previous shifts, are needed. The character = changes the mode of subsequent shifts to absolute, while the character + changes the mode of subsequent shifts to incremental.

The character ! sets the accumulated incremental shift to 0.

The mode and accumulated incremental shift are state variables.

## Saving and Restoring the State

Since row and column transformations may be different, separate states are kept for each.

The symbol [ saves the current state and the symbol ] restores it.

For example,

~1[=66]1

complements the first line, shifts the next two lines incrementally by 1 (corresponding to absolute shifts of 1 and 2), saves the state, sets the shift mode to absolute, shifts the next two lines by 6, restores the mode and accumulated incremental shift, and shifts the next line incrementally by 1 (a total of 3).

## Repeats

Normally if the length of the operation sequence (excluding characters that do not perform operations) does not evenly divide the number of lines, the pattern is repeated so that it does [1].

The character : at the end of an operation sequence prevents pattern repeats and terminates line processing when it is encountered; the operation sequence is not repeated .

The character ; at the end of an operation sequence prevents pattern repeats but repeats the operation sequence until the last line is reached.

## Operation Summary

.	no operation
~	complementation
	reversal
<i>n</i>	shift by <i>n</i>
a ... z	shift by alphabetic position + 9
{ ... }	combination operations
( ... )	permutation
>	shift right
<	shift left
=	absolute shift
+	incremental shift
!	zero accumulated incremental shift
:	do not repeat pattern and stop
;	do not repeat pattern but continue

## Next Time

The next article in this series will give examples of line transformations.

## References

1. *Color-Complementation, Part 1: Color-Alternate Weaves*, Ralph E. Griswold, 2004:

[http://www.cs.arizona.edu/patterns/weaving/webdocs/gre\\_com1.pdf](http://www.cs.arizona.edu/patterns/weaving/webdocs/gre_com1.pdf)

2. *Color-Complementation, Part 2: Complementation Sequences*, Ralph E. Griswold, 2004:

[http://www.cs.arizona.edu/patterns/weaving/webdocs/gre\\_com2.pdf](http://www.cs.arizona.edu/patterns/weaving/webdocs/gre_com2.pdf)

Ralph E. Griswold  
Department of Computer Science  
The University of Arizona  
Tucson, Arizona

© 2004 Ralph E. Griswold