SPREADSHEET LOGOLOGY: LETTER-SHIFTS

ANTHONY SEBASTIAN
San Francisco, California

This article describes a personal computer spreadsheet model (template) that aids in searching for alphabetical letter-shift word-pairs (e.g., **cheer-jolly**, shift-value 7). It complements my November 1988 **Word Ways** article demonstrating how non-computer-experts can tap the computer's power for logological studies by using commercially-available, inexpensive spreadsheet applications programs like Lotus 1-2-3. A motivated novice with little experience using personal computers can learn to use a spreadsheet applications program in a few hours.

The letter-shift template is reusable. Once a word is entered into a test cell, the spreadsheet immediately displays a list of letter-sequences for all 25 possible letter-shift values. After scanning the list of legitimate words, the user may enter a new test word into the reusable test cell, whereupon the new word's list of shifted letter-sequences immediately replaces the previous word's list. The template allows for rapid serial testing of large numbers of words; the rate-limiting steps are typing in the test word and scanning the list for mates. The template also can be easily modified to test multiple words at one time, in which case it generates the shifted letter-sequences for the entire set of test words at the same time.

The letter-shift template is an electronic equivalent of David Morice's word calculator, a mechanical device that consists of multiple sets of the alphabet's letters, each set arranged vertically on rotatable parallel strips of paper looped around a horizontal cylinder; for details, the reader is referred to page 119 of the May 1988 **Word Ways.** Rotating the strips relative to one another to spell out a test word across one row of strips simultaneously generates all possible letter-shift mates in the other rows.

The electronic spreadsheet template operates on a somewhat similar principle. It offers the mechanical calculator's flexibility for following up on leads that turn up during a search, with the advantage of greatly speeding up the process, and, if desired, the facility for testing multiple words at one time and producing a permanent printed copy of the results.

Creating a Permanent Look-up Table of Alphabetical Letter-Shift Values

For the letters of the test word, the template generates the letter-shifted letter-sequence for all 25 shift-values by looking up each of the test word's letters in a permanent reference table that

contains a list of the 25 shift-values for each letter of the alphabet.

Like the logological templates described in the previous article, the letter-shift template utilizes an @Function, in this instance @VLOOKUP. A cell containing an @VLOOKUP formula displays a character or character string retrieved from a designated cell of a reusable reference table prepared by the user and permanently stored in an out-of-the-way section of the spreadsheet. That is, an @VLOOKUP formula "looks up" its value in a permanent prefabricated look-up table. The syntax of @VLOOKUP is: @VLOOKUP(string1, tablerange,tablecolumn); it is best explained by an example.

Suppose the role of each of five players in a hypothetical game depends on which of three teams the player is assigned to, as shown in rows 1-7 of the following spreadsheet. What role does player3 play when he is on team2? The answer is given by @VLOOKUP(A10, A3..D7,A11) in cell A12.

```
      A         B        C        D
 1            team1    team2    team3
 2
 3  player1   captain  cook     clerk
 4  player2   cook     clerk    driver
 5  player3   clerk    lawyer   captain
 6  player4   driver   captain  lawyer
 7  player5   lawyer   driver   cook
 8
 9
10  player3   [enter player]
11        2   [enter tablecolumn (= team number)]
12  lawyer    [contains @VLOOKUP(A10,A3..D7,A11)]
```

Here is how that formula was arrived at: in the example, the look-up table proper is a rectangular range of cells extending from A3 (upper left) to D7 (lower right), specified as A3..D7. Thus, the tablerange parameter in the @VLOOKUP formula for this table will be A3..D7. The string1 parameter, entered in A10, is the player whose role is to be looked up, in this case player3. The tablecolumn parameter is specified as a number indicating which column to look up to find that player's role, in this case which team to look up. The tablecolumn number is always specified as the number of columns to the right of the column with the player list. In this example, that number is 2 and is entered in cell A11. Thus:

```
string1 = player3 = A10
tablerange = look-up table proper = A3..D7
tablecolumn = 2 = A11
```

Cells A10 and A11 can be used over and over for any player-team combination, the answer each time displayed in A12. Once the template is built, those are the only two cells the user will ever make entries in to use the template. With a database that contains hundreds of columns and thousands of rows, such a look-up table can be very useful.

The template for generating letter-shift letter-sequences is considerably more complex. The "player list" in the look-up table is the alphabet, one letter per cell in alphabetical order vertically in 26 cells in the left-hand column of the table below. To the right in the table are 25 additional columns, each also with 26 cells containing one letter per cell arranged alphabetically. However, with each successive column the alphabetic listing begins one letter further in the alphabet, so that after the leftmost column, the first column begins with "b", the second with "c", etc. In each column, when "z" is reached, the listing continues with "a", "b", etc., so that each column has all 26 letters.

The result is a look-up table that supplies to the right of each letter of the alphabet a row of letters representing each successive shift value, 1 through 25. We locate the table in the rectangle extending from AA2 to AZ27, so that AA2..AZ27 becomes the table-range parameter used for every @VLOOKUP formula in the template. The first and last six rows and columns of the spreadsheet section containing the look-up table is shown below. The reader can easily deduce the missing rows and columns from the pattern shown.

	AA	AB	AC	AD	AE	AF	.....	AU	AV	AW	AX	AY	AZ
1		1	2	3	4	5	.....	20	21	22	23	24	25
2	a	b	c	d	e	f	.....	u	v	w	x	y	z
3	b	c	d	e	f	g	.....	v	w	x	y	z	a
4	c	d	e	f	g	h	.....	w	x	y	z	a	b
5	d	e	f	g	h	i	.....	x	y	z	a	b	c
6	e	f	g	h	i	j	.....	y	z	a	b	c	d
.	.	.	.	.	.	.		.	.	.	.	.	.
.	.	.	.	.	.	.		.	.	.	.	.	.
22	u	v	w	x	y	z	.....	o	p	q	r	s	t
23	v	w	x	y	z	a	.....	p	q	r	s	t	u
24	w	x	y	z	a	b	.....	q	r	s	t	u	v
25	x	y	z	a	b	c	.....	r	s	t	u	v	w
26	y	z	a	b	c	d	.....	s	t	u	v	w	x
27	z	a	b	c	d	e	.....	t	u	v	w	x	y

Setting Up the Working Area of the Template

A1..R10 will be set aside as the working area of the template, where the letter-shift sequences are generated. For expository convenience, we illustrate a template for five-letter words; the reader will want to build his own template for a longer word length. The template display is given on the following page (using "cheer" as test word).

The display-width has been set to 1 for columns B-F, H-L, N-R, since the cells in those columns will each display only 1 letter, specifically a letter from the look-up table. By setting the display-

width to 1 for these columns, there will be no spaces between the letters in adjacent columns, so as to give the appearance of spelled out words. Cell A1 is the reusalbe cell where test words are entered. The numbers 1 through 25, representing the 25 shift values, are entered in the cells of columns A, G, and M as shown.

The cells of each row of columns B-F, H-L, N-R contain the appropriate @VLOOKUP formulas to generate letter-shifts of a single shift value for the letters of "cheer". (Remember from the preceding discussion of spreadsheet basics that, regardless of a cell's display width, the formula in the cell can be up to 240 characters long.)

	A	BCDEF	G	HIJKL	M	NOPQR
1	cheer					
2	1	diffs	10	mroob	19	vaxxk
3	2	ejggt	11	nsppc	20	wbyyl
4	3	fkhhu	12	otqqd	21	xczzm
5	4	gliiv	13	purre	22	ydaan
6	5	hmjjw	14	qvssf	23	zebbo
7	6	inkkx	15	rwttg	24	afccp
8	7	jolly	16	sxuuh	25	bgddq
9	8	kpmmz	17	tyvvi		
10	9	lqnna	18	uzwwj		

Let us examine what formulas we need to obtain the sequence for shift value = 1 ("diffs") from "cheer". The letters of "diffs" are in the row of cells B2 through F2. The "d" is in B2. To get "d" from "c", we find "c" in the first column of the look-up table, then move 1 column to the right because 1 is the shift value, there finding "d". Similarly, looking up "h" gives "i" 1 column over, the "e"'s give "f"'s, and the "r" gives "s".

For the template to accomplish these look-ups, each @VLOOKUP formula in the cells B2 through F2 must use a different letter of "cheer" for its string1 parameter. The next step, then, is to determine how to specify each letter of "cheer".

To designate a specific letter in a word, we use the @MID function described in the previous article. By way of review, a cell containing a single @MID function displays a specific letter or consecutive sequence of letters from a test word, according to the following syntax: @MID(string1,num1,num2), where num2 specifies how many letters from string1 to display beginning at string1's letter position designated by num1. (Remember, the first letter position in a string is position 0, just as the first number in the number line is 0.) Thus, @MID("cheer",0,1) would display "c", @MID("cheer",1,1) would display "h", @MID("cheer",2,1) would display "e", etc.

With that background, we can determine the formula to enter into each cell. The formulas for the first three shift values are shown on the next page. The remaining formulas in the template can be deduced from the pattern shown.

In these @VLOOKUP(string1,tablerange,tablecolumn) formulas note that (a) the string1 parameter is given by the appropriate @MID

formula that selects the desired letter of the test word; (b) the tablerange parameter is invariant; (c) the tablecolumn parameter is equal to the letter shift value, and is given by the address of the cell containing that number located to the left of each let-shift sequence.

```
CELL B2--> @VLOOKUP(@MID(A1,0,1),AA2..AZ27,A2)
CELL C2--> @VLOOKUP(@MID(A1,1,1),AA2..AZ27,A2)
CELL D2--> @VLOOKUP(@MID(A1,2,1),AA2..AZ27,A2)
CELL E2--> @VLOOKUP(@MID(A1,3,1),AA2..AZ27,A2)
CELL F2--> @VLOOKUP(@MID(A1,4,1),AA2..AZ27,A2)

CELL B3--> @VLOOKUP(@MID(A1,0,1),AA2..AZ27,A3)
CELL C3--> @VLOOKUP(@MID(A1,1,1),AA2..AZ27,A3)
CELL D3--> @VLOOKUP(@MID(A1,2,1),AA2..AZ27,A3)
CELL E3--> @VLOOKUP(@MID(A1,3,1),AA2..AZ27,A3)
CELL F3--> @VLOOKUP(@MID(A1,4,1),AA2..AZ27,A3)

CELL B4--> @VLOOKUP(@MID(A1,0,1),AA2..AZ27,A4)
CELL C4--> @VLOOKUP(@MID(A1,1,1),AA2..AZ27,A4)
CELL D4--> @VLOOKUP(@MID(A1,2,1),AA2..AZ27,A4)
CELL E4--> @VLOOKUP(@MID(A1,3,1),AA2..AZ27,A4)
CELL F4--> @VLOOKUP(@MID(A1,4,1),AA2..AZ27,A4)
```

The template works for five-letter words as well as for words up to five letters. With a four-letter test word, an asterisk appears in the final letter position for each shift-value sequence. Similarly, two final asterisks appear when a three-letter test word is entered. The asterisks are simply ignored when the user scans the list for mates of the test word. The reader will probably want to construct his own template for a longer word length than five letters; the principles are the same, and the product more useful.

There are several ways to modify the template to operate on a list of test words simultaneously. The most straightforward approach is to arrange the formulas for each test word so that its 25 letter-shift sequences are displayed in a single column-set instead of in three column-sets as shown in the example above; the number of columns in each column-set depends on the letter-count of the test words (e.g., 5 columns per column-set for 5-letter test words). Remember to set the display width to 1 for each column of the set. In parallel columns across the available width of the spreadsheet, place a column-set for each of the test words. Reserve one column to hold the list of test words itself, and 26 columns for the look-up table. In the cell formulas of each of the letter-shift columns, refer to the address set aside for the column's test word. Then, each time a new list of test words is entered into the test-word column, the letter-shift sequences for all the test words display simultaneously. A sample display for a ten-word list is shown on the next page.

Significance of the Spreadsheet Approach to Letter-Shift Generation

The spreadsheet approach to generation of letter-shift word-pairs is not intended to be a definitive computer solution. A computer

LETTER-SHIFT SEQUENCES FOR A TEN-WORD LIST

SHIFT VALUE --->											
	1	bbbbbbb	ccccccc	ddddddd	eeeeeee	fffffff	ggggggg	hhhhhhh	iiiiiii	jjjjjjj	kkkkkkk
	2	ccccccc	ddddddd	eeeeeee	fffffff	ggggggg	hhhhhhh	iiiiiii	jjjjjjj	kkkkkkk	lllllll
	3	ddddddd	eeeeeee	fffffff	ggggggg	hhhhhhh	iiiiiii	jjjjjjj	kkkkkkk	lllllll	mmmmmmm
TESTLIST	4	eeeeeee	fffffff	ggggggg	hhhhhhh	iiiiiii	jjjjjjj	kkkkkkk	lllllll	mmmmmmm	nnnnnnn
aaaaaaa	5	fffffff	ggggggg	hhhhhhh	iiiiiii	jjjjjjj	kkkkkkk	lllllll	mmmmmmm	nnnnnnn	ooooooo
bbbbbbb	6	ggggggg	hhhhhhh	iiiiiii	jjjjjjj	kkkkkkk	lllllll	mmmmmmm	nnnnnnn	ooooooo	ppppppp
ccccccc	7	hhhhhhh	iiiiiii	jjjjjjj	kkkkkkk	lllllll	mmmmmmm	nnnnnnn	ooooooo	ppppppp	qqqqqqq
ddddddd	8	iiiiiii	jjjjjjj	kkkkkkk	lllllll	mmmmmmm	nnnnnnn	ooooooo	ppppppp	qqqqqqq	rrrrrrr
eeeeeee	9	jjjjjjj	kkkkkkk	lllllll	mmmmmmm	nnnnnnn	ooooooo	ppppppp	qqqqqqq	rrrrrrr	sssssss
fffffff	10	kkkkkkk	lllllll	mmmmmmm	nnnnnnn	ooooooo	ppppppp	qqqqqqq	rrrrrrr	sssssss	ttttttt
ggggggg	11	lllllll	mmmmmmm	nnnnnnn	ooooooo	ppppppp	qqqqqqq	rrrrrrr	sssssss	ttttttt	uuuuuuu
hhhhhhh	12	mmmmmmm	nnnnnnn	ooooooo	ppppppp	qqqqqqq	rrrrrrr	sssssss	ttttttt	uuuuuuu	vvvvvvv
iiiiiii	13	nnnnnnn	ooooooo	ppppppp	qqqqqqq	rrrrrrr	sssssss	ttttttt	uuuuuuu	vvvvvvv	wwwwwww
jjjjjjj	14	ooooooo	ppppppp	qqqqqqq	rrrrrrr	sssssss	ttttttt	uuuuuuu	vvvvvvv	wwwwwww	xxxxxxx
	15	ppppppp	qqqqqqq	rrrrrrr	sssssss	ttttttt	uuuuuuu	vvvvvvv	wwwwwww	xxxxxxx	yyyyyyy
	16	qqqqqqq	rrrrrrr	sssssss	ttttttt	uuuuuuu	vvvvvvv	wwwwwww	xxxxxxx	yyyyyyy	zzzzzzz
	17	rrrrrrr	sssssss	ttttttt	uuuuuuu	vvvvvvv	wwwwwww	xxxxxxx	yyyyyyy	zzzzzzz	aaaaaaa
	18	sssssss	ttttttt	uuuuuuu	vvvvvvv	wwwwwww	xxxxxxx	yyyyyyy	zzzzzzz	aaaaaaa	bbbbbbb
	19	ttttttt	uuuuuuu	vvvvvvv	wwwwwww	xxxxxxx	yyyyyyy	zzzzzzz	aaaaaaa	bbbbbbb	ccccccc
	20	uuuuuuu	vvvvvvv	wwwwwww	xxxxxxx	yyyyyyy	zzzzzzz	aaaaaaa	bbbbbbb	ccccccc	ddddddd
	21	vvvvvvv	wwwwwww	xxxxxxx	yyyyyyy	zzzzzzz	aaaaaaa	bbbbbbb	ccccccc	ddddddd	eeeeeee
	22	wwwwwww	xxxxxxx	yyyyyyy	zzzzzzz	aaaaaaa	bbbbbbb	ccccccc	ddddddd	eeeeeee	fffffff
	23	xxxxxxx	yyyyyyy	zzzzzzz	aaaaaaa	bbbbbbb	ccccccc	ddddddd	eeeeeee	fffffff	ggggggg
	24	yyyyyyy	zzzzzzz	aaaaaaa	bbbbbbb	ccccccc	ddddddd	eeeeeee	fffffff	ggggggg	hhhhhhh
	25	zzzzzzz	aaaaaaa	bbbbbbb	ccccccc	ddddddd	eeeeeee	fffffff	ggggggg	hhhhhhh	iiiiiii

programmer with access to a powerful computer and extensive computer-readable lexicon could, without difficulty, write a specialized program to generate all of the lexicon's legitimate letter-shift word-pairs (or word-sets if more than a pair exists for any word). Such compendia would undoubtedly be of great value to logology.

Still, there will always be other word lists, including specialty dictionaries (law, medicine, music, etc.), and new words appearing that the logologist who is not a computer expert will want to examine. The personal computer spreadsheet approach described in this article eliminates the logologist's complete dependence on the computer programmer. It gives the non-computer-expert a tool to facilitate what would otherwise be a slow, tedious and error-prone process, enabling timely productivity and preserving the joy of personal discovery.