An Algebraic Investigation of the APL "Grade-Up" Function

Curtis N. Cooper, Robert E. Kennedy, Clarence H. Best
Central Missouri State University
Warrensburg, Missouri

APL is a powerful computer language which derives its elegance from its versatility in handling arrays. Its highly symbolic notation makes it a language especially suitable for mathematical and scientific programming. It is hard not to appreciate a language which enables the programmer to write shorter programs for the solution of problems than would be required using other languages.

Here, we do not intend to go into detail about APL. In fact, no knowledge of APL is needed to understand what follows. The purpose of this article is to investigate the so-called "grade-up" function of APL from an algebraic viewpoint. Although we have tried to make what follows as self-contained as possible, the references given at the end of this article will include the necessary background needed with respect to algebra and the APL language.

The grade-up function, $\uparrow V$, outputs the relative positions of the elements of a vector (ordered n-tuple), V, in ascending order. That is, if the ith coordinate of a vector V is $V(i)$, then $\uparrow V$ returns a vector I consisting of some rearrangement of the indices of V, that would order V, in the sense that

$$V[I] = V[\uparrow V]$$

has its elements in increasing order. Thus one use of the grade-up function is to sort a list. For example, suppose that V is a list of numbers. Then the APL statement

$$V[\uparrow V]$$

outputs the elements of V in ascending order. For a specific example, let $V = (5, 4, 6, 9, 3)$. Then the position of the smallest component (which is 3)

Curtis Cooper received his education at Culver-Stockton College and Iowa State University. He enjoys tennis, softball, and bowling.
Robert E. Kennedy has been teaching for 10 years. His other interests include science fiction, coin shooting, ornithology, skiing, bowling and golf.
Clarence H. Best enjoys being a spectator of football and baseball, as well as his hobbies of numismatics, philately and Bluegrass music.

is in position 5, the position of the next smallest component (which is 4) is in position 2, and so forth. Thus

$$\uparrow V = (5, 2, 1, 3, 4).$$

If it happens that V has duplicate components, then the grade-up function outputs the indices of the duplicates in ascending order. So, for example,

$$\uparrow (2, 3, 2, 1, 5,) = (4, 1, 3, 2, 5).$$

The motivation for what follows was the problem, "characterize all vectors V such that $\uparrow V = V$", together with the desire to use the algebraic theory of permutations to mathematically characterize the action of the grade-up function.

Let S_n be the set of permutations on the set of positive integers less than or equal to n, and let R^n denote the collection of ordered n-tuples over the real numbers. We define the mapping $F\colon S_n \to R^n$ by

$$F(s) = (s(1), s(2), s(3), \ldots, s(n)), \text{ for } s \epsilon S_n.$$

Since each $s \epsilon S_n$ is a one-to-one mapping, it is immediate that F is also a one-to-one mapping.

For each $s \epsilon S_n$, define

$$P_s = \{(x_1, x_2, \ldots, x_n) \epsilon R^n \mid x_{s(j)} \sim x_{s(j+1)}, \text{ for } j = 1, 2, \ldots, n-1, \text{ where}$$
$$\sim \text{ is replaced by}$$
$$\leqslant \text{ if } s(j) < s(j+1) \text{ and is replaced by } < \text{ if } s(j) > s(j+1)\}.$$

To exemplify the concept of P_s, suppose that $n = 4$ and $s \epsilon S_4$ where $s(1) = 3$, $s(2) = 1$, $s(3) = 2$ and $s(4) = 4$. Then

$$P_s = \{(x_1, x_2, x_3, x_4) \epsilon R^4 \mid x_3 < x_1 \leqslant x_2 \leqslant x_4\}.$$

Hence, the elements of P_s have their elements ordered by their permuted indices. Also note that the strict inequality forces each P_s to be disjoint from the other P_s's. Thus every collection of n real numbers can be sorted in ascending order with respect to $\sim$ in exactly one way and so, we have the following theorem

Theorem 1. *The collection $\{P_s \mid s \epsilon S_n\}$ is a partition of R^n for each positive integer n.*

As an example of how one may determine the partition class to which a particular V belongs, consider $V = (1.5, 2.0, -3) \epsilon R^4$. Then $V \epsilon P_s$ where $s(1) = 4$, $s(2) = 3$, $s(3) = 1$, and $s(4) = 2$.

If $V \epsilon R^n$, then $V \epsilon P_s$ for exactly one $s \epsilon S_n$ by the above theorem. In addition, if $V = (x_1, x_2, \ldots, x_n)$, then the list of components $x_1, x_2, \ldots, x_n$ in ascending order is

$$x_{s(1)}, x_{s(2)}, \ldots, x_{s(n)}, \text{ for some } s \epsilon S_n.$$

This leads to the following definition which corresponds to the APL definition of $\uparrow V$.

Definition. *Define the mapping*

$$\uparrow: R^n \to R^n$$

$$\textit{by } \uparrow V = F(s) \textit{ where } V \epsilon P_s, \textit{ for } s \epsilon S_n.$$

The next theorem describes what the grade-up function does to the vector $F(s)$. Recall that when considering S_n as a multiplicative group, s^{-1} denotes the multiplicative inverse of s.

Theorem 2. $\uparrow F(s) = F(s^{-1})$ *for each* $s \epsilon S_n$.

Proof Note that

$$\uparrow F(s) = \uparrow (s(1), s(2), \ldots, s(n)) = (i_1, i_2, \ldots, i_n)$$

where $s(i_1) = 1, s(i_2) = 2, \ldots, s(i_n) = n$. Since for each $m = 1, 2, \ldots, n$, we have that $s^{-1}(s(i_m)) = i_m$, it follows that $s^{-1}(m) = i_m$ for $m = 1, 2, \ldots, n$. Thus $\uparrow F(s) = F(s^{-1})$ since

$$F(s^{-1}) = (s^{-1}(1), s^{-1}(2), \ldots, s^{-1}(n)) = (i_1, i_2, \ldots, i_n).$$

From *Theorems 1 and 2*, we have the following corollaries.

Corollary. *For any* $V \epsilon R^n$, $\uparrow\uparrow\uparrow V = \uparrow V$ *where* $\uparrow\uparrow\uparrow V$ *denotes* $\uparrow(\uparrow(\uparrow V))$.

Proof For $s \epsilon S_n$, let $V \epsilon P_s$. The proof of the corollary follows since

$$\uparrow\uparrow\uparrow V = \uparrow\uparrow F(s) = F(s^{-1}) = F((s^{-1})^{-1}) = F(s) = \uparrow V.$$

Corollary. *Let* $s \epsilon S_n$. *Then* $\uparrow F(s) = F(s)$ *if and only if* $s = s^{-1}$ (*that is*, s^2 *is the identity permutation*).

Proof If $s = s^{-1}$, then *Theorem 2* gives that $\uparrow F(s) = F(s)$. If, on the other hand, $\uparrow F(s) = F(s)$, it follows from *Theorem 2* that $F(s^{-1}) = F(s)$. But then $s^{-1} = s$ since F is a one-to-one mapping.

The following result answers the question of when $\uparrow V = V$.

Theorem 3. *Let $V \epsilon P_s$ for $s \epsilon S_n$. Then $\uparrow V = V$ if and only if $V = F(s)$ and $s = s^{-1}$.*

Proof If $V = F(s)$ and $s = s^{-1}$, then it follows from the above corollary that $\uparrow V = V$. If $\uparrow V = V$, then by the definition of $\uparrow$, $V = F(s)$, and again by the above corollary, $s = s^{-1}$.

As a demonstration of *Theorem 3*, let $s \epsilon S_4$ where $s(1) = 2$, $s(2) = 1$, $s(3) = 4$ and $s(4) = 3$. Here $s = s^{-1}$ and $F(s) = (2, 1, 4, 3)$. Thus if $V = (2, 1, 4, 3)$, V is such that $\uparrow V = V$.

The preceding theorems and corollaries were intended to demonstrate how it is possible to investigate a particular APL function from an algebraic standpoint. Other such functions can also be studied in a similar way. For example, you might wish to investigate the "grade-down" APL function in this same spirit.

The grade-down function, $\downarrow V$, outputs the relative positions of the elements of a vector, V, in *descending* order. One way to investigate this function is to realize that $\downarrow V = \uparrow (-V)$ where $-V$ represents the vector found by replacing each component of V by its negative. Using this, we would like to conclude this paper by presenting some questions which you might wish to consider.

1. Find conditions on a vector V such that $\downarrow V = V$.
2. Prove or disprove. If $V \epsilon R^n$,

$$\text{(a) } \downarrow\downarrow\downarrow\downarrow\downarrow V = \downarrow V.$$

$$\text{(b) } \downarrow\uparrow\downarrow\uparrow\downarrow V = \downarrow V.$$

REFERENCES

1 Birkhoff, G.; MacLane, S. *A Survey of Modern Algebra*, New York: The MacMillan Company, 1965.

2 Gilman, L.; Rose, A. *APL An Interactive Approach*, New York: John Wiley & Sons, Inc., 1976.

3 IBM Manual, *APL Language*, International Business Machines, 1978.