Multiple passes through a stack

Rebecca Smith (SUNY Brockport)

Knuth [1] showed that a permutation π can be sorted by a stack (meaning that by applying push and pop operations to the sequence of entries $\pi(1), \ldots, \pi(n)$ we can output the sequence $1, \ldots, n$) if and only if π avoids the permutation 231, i.e., if and only if there do not exist three indices $1 \leq i_1 < i_2 < i_3 \leq n$ such that $\pi(i_1), \pi(i_2), \pi(i_3)$ are in the same relative order as 231.

We consider the number of passes a permutation needs to take through a stack if we only pop the appropriate output values and start over with the remaining entries. We define a permutation π to be k-pass sortable if π is sortable using k passes through the stack where the remaining entries are passed through the stack in their original order on later attempts. Permutations that are 1-pass sortable are simply the stack sortable permutations as defined by Knuth. We define the permutation class of 2-pass sortable permutations in terms of their basis.

Example 1. *We show the sorting of* $\pi = 356124$ *using three passes through a stack in Figure 1.*

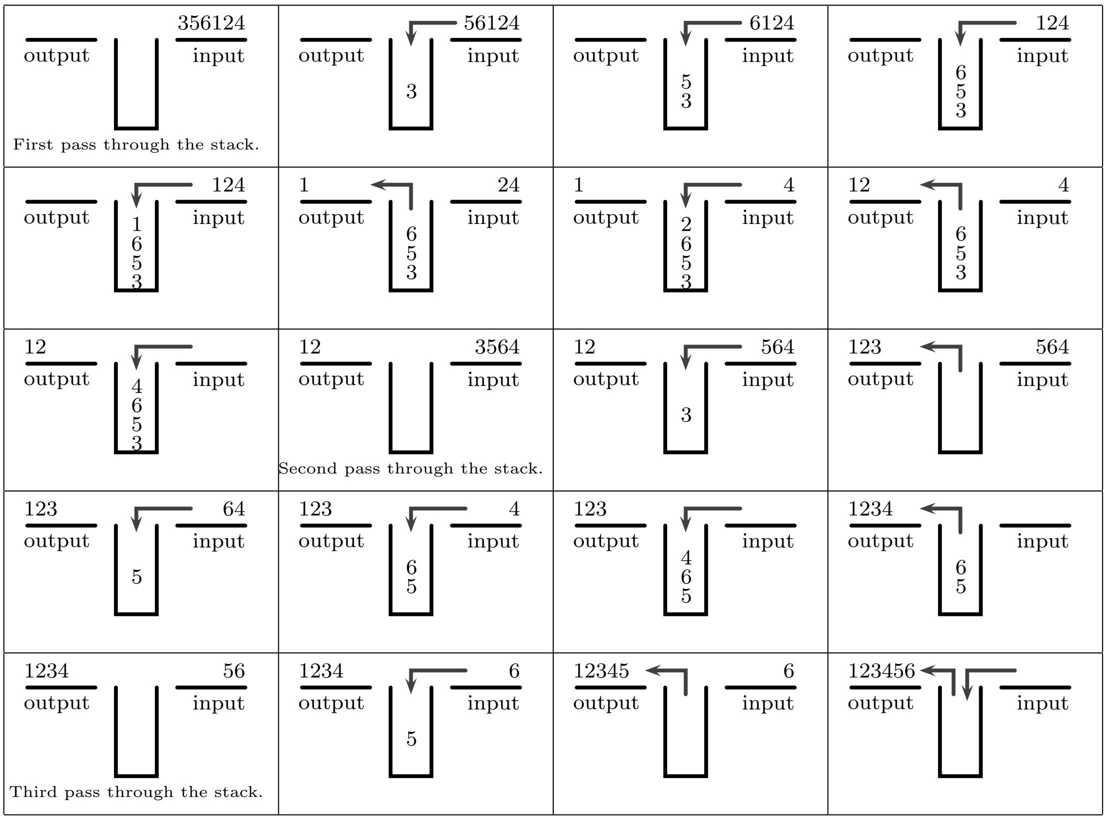

Figure 1: Sorting the permutation 356124 with $k = 3$ passes through a stack.

We define the notion of tier of a permutation π to be the minimum number of passes *after* the first pass required to sort π. We then give a bijection between the class of permutations of tier t and a collection of integer sequences studied by Parker [2]. This gives an exact enumeration of tier t permutations of a given length and thus an exact enumeration for the class of $(t+1)$-pass sortable permutations. This also allows us to give a new derivation for the generating function in [2] and an explicit formula for the coefficients.

Finally we give some results on reverse k-pass-sortable permutations where outputting the appropriate values is still the priority, but then remaining entries are processed in the reverse order from the previous pass.

This talk is based on joint work with Toufik Mansour and Howard Skogman.

References

[1] Knuth, D. E. *The art of computer programming. Volume 1.* Addison-Wesley Publishing Co., Reading, Mass., 1969. Fundamental Algorithms.

[2] Parker, S. *The Combinatorics of Functional Composition and Inversion.* PhD thesis, Brandeis U., 1993.