

Wednesday 4/30

The RSK Correspondence

Definition 1. Let $\lambda \vdash n$. A *standard [Young] tableau of shape λ* is a filling of the Ferrers diagram of λ with the numbers $1, 2, \dots, n$ that is increasing left-to-right and top-to-bottom.

We write $SYT(\lambda)$ for the set of all standard tableaux of shape λ , and set

$$f^\lambda = |SYT(\lambda)|.$$

For example, if $\lambda = (3, 3)$, then $f^\lambda = 5$; the members of $SYT(\lambda)$ are as follows:

1	3	5	1	3	4	1	2	5	1	2	4	1	2	3
2	4	6	2	5	6	3	4	6	3	5	6	4	5	6

The *RSK correspondence* (for Robinson-Schensted-Knuth) constructs, for every permutation $w \in \mathfrak{S}_n$, a pair $RSK(w) = (P, Q)$ of standard tableaux of the same shape $\lambda \vdash n$. The idea is to “row-insert” the numbers w_1, w_2, \dots, w_n into P one by one, and to use Q to record the order in which cells are added.

Example 1. Let $w = 57214836 \in \mathfrak{S}_8$. We start with a pair (P, Q) of empty tableaux.

Step 1: Row-insert $w_1 = 5$ into P . We do this in the obvious way. Since it’s the first cell added, we add a cell containing 1 to Q .

$$(1a) \quad P = \boxed{5} \quad Q = \boxed{1}$$

Step 2: Row-insert $w_2 = 7$ into P . Since $5 < 7$, we can do this by appending the new cell to the top row, and adding a cell labeled 2 to Q to record where we’ve put the new cell in P .

$$(1b) \quad P = \boxed{5 \ 7} \quad Q = \boxed{1 \ 2}$$

Step 3: Row-insert $w_3 = 2$ into P . This is a bit trickier. We can’t just append a 2 to the first row of P , because the result would not be a standard tableau. The 2 has to go in the top left cell, but that already contains a 5. Therefore, the 2 “bumps” the 5 out of the first row into a new second row. Again, we record the location of the new cell by adding a cell labeled 3 to Q .

$$(1c) \quad \begin{array}{|c|c|} \hline 2 & 7 \\ \hline 5 & \\ \hline \end{array} \quad \begin{array}{|c|c|} \hline 1 & 2 \\ \hline 3 & \\ \hline \end{array}$$

Step 4: Row-insert $w_4 = 1$ into P . This time, the new 1 bumps the 2 out of the first row. The 2 has to go into the second row, but again we can’t simply append it to the right. Instead, the 2 bumps the 5 out of the second row into the (new) third row.

(1d)

1	7
2	
5	

1	2
3	
4	

Step 5: Row-insert $w_5 = 4$ into P . The 4 bumps the 7 out of the first row. The 7, however, can comfortably fit at the end of the second row, without any more bumping.

(1e)

1	4
2	7
5	

1	2
3	5
4	

Step 6: Row-insert $w_6 = 8$ into P . The 8 just goes at the end of the first row.

(1f)

1	4	8
2	7	
5		

1	2	6
3	5	
4		

Step 7: Row-insert $w_7 = 3$ into P . 3 bumps 4, and then 4 bumps 7.

(1g)

1	3	8
2	4	
5	7	

1	2	6
3	5	
4	7	

Step 8: Row-insert $w_8 = 6$ into P . 6 bumps 8 into the second row.

(1h)

1	3	6
2	4	8
5	7	

1	2	6
3	5	8
4	7	

A crucial feature of the RSK correspondence is that it can be reversed. That is, given a pair (P, Q) , we can recover the permutation that gave rise to it.

Example 2. Suppose that we were given the pair of tableaux in (1h). What was the previous step? To get the previous Q , we just delete the 8. As for P , the last cell added must be the one containing 8. This is in the second row, so somebody must have bumped 8 out of the first row. That somebody must be the largest number less than 8, namely 6. So 6 must have been the number inserted at this stage, and the previous pair of tableaux must have been those in (1g).

Example 3. Suppose P is the standard tableau with 18 boxes shown on the left.

1	2	5	8	10	18
3	4	11	12		
6	7	13			
9	15	17			
14	16				

1	2	5	8	10	18
3	4	11	12		
6	7	13			
9	15	17			
14	16				

Suppose in addition that we know that the cell labeled 16 was the last one added (because the corresponding cell in Q contains an 18). Then the “bumping path” must be as shown on the right. (That is, the 16 was bumped by the 15, which was bumped by the 13, and so on.) To find the previous tableau in the algorithm, we push every number in the bumping path up and toss out the top one.

1	2	5	8	12	18
3	4	11	13		
6	7	15			
9	16	17			
14					

That is, we must have gotten the original tableau by row-inserting 10 into the tableau just shown.

Since the correspondence is reversible, we have the following fact:

Theorem 1. *The RSK correspondence is a bijection*

$$\mathfrak{S}_n \xrightarrow{RSK} \bigcup_{\lambda \vdash n} SYT(\lambda) \times SYT(\lambda).$$

Some Nice Properties of RSK

Here is an immediate consequence of Theorem 1:

Corollary 2. $\sum_{\lambda \vdash n} (f^\lambda)^2 = n!$.

To confirm this for $n = 3$, here are all the SYT’s with 3 boxes:

1	2	3
---	---	---

1	2
3	

1	3
2	

1
2
3

Note that $f^{(3)} = f^{(1,1,1)} = 1$ and $f^{(2,1)} = 2$, and $1^2 + 1^2 + 2^2 = 6 = 3!$. This calculation ought to look familiar!

Another neat fact about the RSK correspondence is this:

Proposition 3. *Let $w \in \mathfrak{S}_n$. If $RSK(w) = (P, Q)$, then $RSK(w^{-1}) = (Q, P)$.*

In particular, the number of involutions in \mathfrak{S}_n is $\sum_{\lambda \vdash n} f^\lambda$.

Generalized RSK and Schur Functions

The RSK correspondence can be extended to obtain more general tableaux than just SYT's. We can think of a permutation in two-line notation, i.e.,

$$57214836 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 5 & 7 & 2 & 1 & 4 & 8 & 3 & 6 \end{pmatrix}.$$

What if we allowed “generalized permutations”, i.e., things of the form

$$(2) \quad w = \begin{pmatrix} \underline{a} \\ \underline{b} \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & \cdots & a_n \\ b_1 & b_2 & \cdots & b_n \end{pmatrix}$$

where $a_i, b_i \in [n]$, and the ordered pairs $(a_1, b_1), \dots, (a_n, b_n)$ are in lexicographic order, but repeats are allowed?

Example 4. Consider the generalized permutation

$$w = \begin{pmatrix} 1 & 1 & 2 & 4 & 4 & 4 & 5 & 5 & 5 \\ 2 & 4 & 1 & 1 & 3 & 3 & 2 & 2 & 4 \end{pmatrix}.$$

We can row-insert the elements of the bottom row into a tableau P while recording the elements of the top row in a tableau Q :

$$\begin{array}{cc} \mathbf{P} = & \boxed{1} & \mathbf{Q} = & \boxed{1} \\ & & & \\ & \boxed{2 \mid 4} & & \boxed{1 \mid 1} \\ & & & \\ & \boxed{1 \mid 4} & & \boxed{1 \mid 1} \\ & \boxed{2} & & \boxed{2} \\ & & & \\ & \boxed{1 \mid 1} & & \boxed{1 \mid 1} \\ & \boxed{2 \mid 4} & & \boxed{2 \mid 4} \end{array}$$

etc.

The tableaux P, Q we get in this way will always be weakly increasing eastward and strictly increasing southward—that is, they will be *column-strict tableaux*. Moreover, their weights will be

$$x^P = x_{a_1} \cdots x_{a_n}, \quad x^Q = x_{b_1} \cdots x_{b_n}.$$

Meanwhile, a generalized permutation w as in (2) can be encoded by an $n \times n$ matrix $M = [m_{ij}] \in \mathbb{N}^{n \times n}$ with

$$m_{ij} = \#\{h \mid a_h = i, b_h = j\}.$$

For example, the generalized permutation w of Example 4 corresponds to the integer matrix

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 \end{bmatrix}.$$

Therefore,

$$\begin{aligned}
\prod_{i,j \geq 1} \frac{1}{1 - x_i y_j} &= \sum_{M=[m_{ij}] \in \mathbb{N}^{n \times n}} \prod_{i,j} (x_i y_j)^{m_{ij}} \\
&= \sum_{n \in \mathbb{N}} \sum_{w = \begin{pmatrix} a \\ b \end{pmatrix}} x_{a_1} \cdots x_{a_n} y_{b_1} \cdots y_{b_n} \\
&= \sum_{\lambda} \sum_{P, Q \in SYT(\lambda)} x^P y^Q \\
&= \sum_{\lambda} \left(\sum_{P \in SYT(\lambda)} x^P \right) \left(\sum_{Q \in SYT(\lambda)} y^Q \right) = \sum_{\lambda} s_{\lambda}(x) s_{\lambda}(y).
\end{aligned}$$

Corollary 4. *The Schur functions form an orthonormal \mathbb{Z} -basis for Λ under the Hall inner product.*