University of California, San Diego

Department of Mathematics

Combinatorics of Young Tableaux and its Applications

Author: Ricardo Arruda

Advisor: Steven V. Sam

June 2022

Contents

1 Introduction

In this report, we explore the combinatorics of the young tableau. The young tableau is an inoffensive combinatorial object introduced by Albert Young around the first years of the 20th century. Since its introduction, it has grown into a large body of work, with deep connections to the study of permutations, set partitions, symmetric functions, asymptotic theory of random systems, and representation theory of the symmetric group, just to name a few. Throughout this report, we hope to highlight the importance of this combinatorial object, alongside with all the intricate algorithms related to it.

This report is divided into 3 main sections. In the preliminary section 2, we introduce and develop the necessary notation to present our results in the remaining sections. After this introduction, we have 2 main sections.

In 3, we introduce the famous RSK algorithm, first presented by Schensted and later generalized by Donald Knuth. Using said algorithm, we then prove combinatorial identities regarding the study of permutations and set partitions. At the end of the 3.2, we present a short discussion on arriving at an analogous result for r-crossing of partitions originally posed in [4].

In 4, we connect the study of the tableau to a seamless unrelated subject: the theory of symmetric polynomials and symmetric functions. Throughout this section, we intend to highlight how the combinatorics of the tableau offer shorter and more intuitive proofs to results already known.

We intend that this report is accessible for a variety of audiences, including aspiring mathematicians. We sincerely hope that this report may elucidate how this particular piece of mathematics is useful in solving a wide variety of problems.

2 Preliminaries

In this section, we introduce the notation used throughout this paper. It is important to note that the notation concerning these topics vary widely throughout the literature. Hereafter, we will be consistent with the treatment and notation of this topic presented in Fulton's [1]. In my experience, the following choice of notation and terminology is also the most commonly adopted one by the relevant literature.

2.1 Integer partitions

We call a weakly increasing finite sequence of non-negative integers a **partition**. A partition may be represented as $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$, for some $n \in \mathbb{N}$, $\lambda_i \in \mathbb{Z}_+$ such that $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$. Another useful representation of a partition is $\lambda = (d_1^{a_1}, d_2^{a_2}, \ldots, d_n^{a_n})$. The latter describes the partition with a_i copies of the integer d_i, where $1 \leq i \leq n$. We define the weight of a partition to be $|\lambda| := \sum_{i=1}^{n} \lambda_i$. To the latter, we say that λ *is a partition of* m, or that λ *partitions* m whenever $|\lambda| := \sum_{i=1}^{n} \lambda_i = m$. This is shorthanded by the symbol $\lambda \vdash m$. The *length* of a partition λ is defined to be its number of components λ_i and we denote it by $l(\lambda)$. We shall denote by δ_n the staircase partition of size n. That is,

$$\delta^{(n)} := (n-1, n-2, \cdots, 1, 0)$$

It is often common and useful to identify finite partitions with infinite sequences of numbers where we annex infinitely many zeroes after the last positive number to appear in the original finite partition. For instance, under this identification $\lambda = (\lambda_1, \lambda_2, ..., \lambda_n) \mapsto \lambda^* = (\lambda_1, \lambda_2, ..., \lambda_n, 0, 0, ...)$. The above definitions of weight and length of a partition are applied in this scenario as one would expect. The symbols Υ_n and Υ shall respectively denote the space of all partitions of length n and the space of all infinite partitions with finitely many nonzero entries. Thus, the map described above can be seen as an inclusion map from Υ_n into Υ.

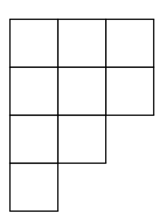
For convenience, wherein the context is appropriate, the empty partition $(0, 0, \ldots)$ shall be denoted by $\emptyset$. We define the addition of two partitions component-wise. That is, if $\lambda = (\lambda_1, \ldots, \lambda_n$, $\rho = (\rho_1, \ldots, \rho_m)$ and WLOG $m \geq n$

$$\lambda + \rho = (\lambda_1 + \rho_1, \ldots, \lambda_n + \rho_n, \rho_{n+1}, \ldots, \rho_m)$$

This operation makes $(\Upsilon_n, +)$ and $(\Upsilon, +)$ into groups. Through the inclusion map, we can identify Υ_n as a subset of Υ. In this case, $(\Upsilon_n, +)$ is a subgroup of $(\Upsilon, +)$.

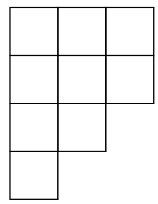
2.2 Young Diagrams

A **Young Diagram** is a collection of boxes arranged from left to right, with a weakly decreasing sequence of boxes in each row. For example,



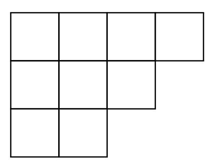
is a Young Diagram.

The definition of a **Young Diagram** suggests that there is a natural correspondence between this object and partitions. Indeed, we can think of them interchangeably: Every partition has a unique young diagram which represents it. That is, the partition $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_n)$ is represented by the Young Diagram which has λ_1 boxes in the first row, λ_2 boxes in the second row, and, inductively, λ_n boxes in the n^{th} row. As an example, the partition $\rho = (3, 3, 2, 1)$ can be realized as

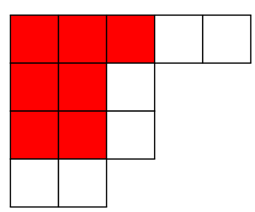
$\rho =$ 

The choice of empty boxes instead of dots or lines is not completely unjustified. We shall shed more light into this when we talk about Young Tableaux. Young Diagrams are sometimes referred to as *Ferrers Diagrams* in the literature as well. Also, Young Diagrams can also be presented upside in the literature; We will mainly use the previous representation, but, when suitable, will shift gears to work with upside-down representation of these.

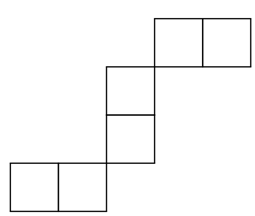
We define the conjugate partition of $\lambda = (\lambda_1, \dots, \lambda_n)$ to be $\lambda' := (\lambda'_1, \dots, \lambda'_n)$, where $\lambda'_i := |\{\lambda_j : \lambda_j \geq i\}|$. Visually, the conjugate partition is obtained by exchanging columns and rows. As an example, if ρ is as above, then

$\rho' =$ 

Now, we introduce the partial ordering $\subset$ on Υ. We declare $\rho \subset \lambda$, whenever $\rho_i \leq \lambda_i$, for all $i \in \mathbb{N}$. This makes $(\Upsilon, \subset)$ into a poset. Visually, we see that $\subset$ merely means that a partition *fits* inside of another. Consider $\rho = (3, 2, 2) \subset (5, 3, 3, 1) = \lambda$. Visually the red boxes represent the young diagram of ρ sitting inside the bigger diagram of λ:



Whenever the relation $\rho \subset \lambda$ is established, we can may define the **skew diagram** ρ/λ. The latter is obtained by removing ρ from λ. If ρ and λ are as in our last example, then

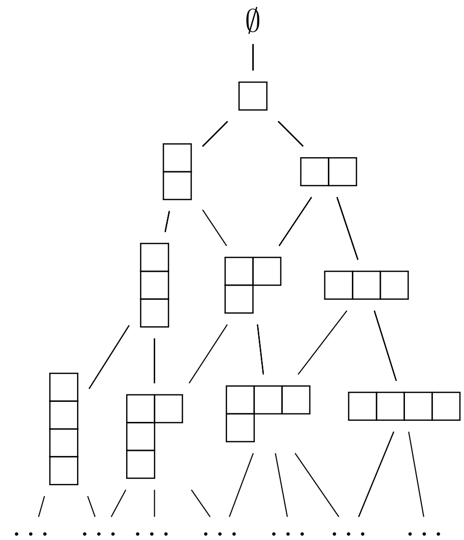
$\rho/\lambda =$ 

The latter is always defined when $\rho \subset \lambda$ but the skew diagram maybe be disconnected. We shall say more about this in the future.

We will now introduce two other partial orderings on Υ, $\leq$ (lexicographic order) and $\ll$ (dominance order). Let $\rho = (\rho_1, \rho_2, \cdots), \lambda = (\lambda_1, \lambda_2, \cdots) \in \Upsilon$. We declare $\rho \leq \lambda$, whenever there exists an i such that $\lambda_i \neq \rho_i$, the first i such that this holds is such that $\rho_i < \lambda_i$. We declare $\rho \ll \lambda$, whenever $\rho_i \leq \lambda_i$ for all i. These two orderings make Υ into a poset, and one can easily verify that

$$\rho \subset \lambda \implies \rho \leq \lambda \implies \rho \ll \lambda$$

Lastly, we will introduce the Young's Lattice, since we will need it in 3.3. The young's lattice is nothing more than the Hasse Diagram for the poset $(\Upsilon, \subset)$. Visually, at the n^{th} branch of the Hasse Diagram, you are dealing with all the possible integer partitions $\lambda \in \Upsilon$ such that $\lambda \vdash n$. The following is a picture:



2.3 Young Tableaux

Any way of arranging positive integers integer in each box of the diagram is called a *filling*. We define a **Young Tableau** as Young Diagram *together* with a filling of the boxes that follows two rules:

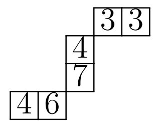
1) weakly increasing across each row
2) strictly increasing down each column

We can also refer to Young diagrams which have arbitrary integers fillings as **tableaux**, however, throughout this report, they will not be of major concern, besides in section 3.2, wherein we talk about permutations. Thus the word **tableau** refers to a Young Tableau, and, should we talk about tableaux with arbitrary fillings, the context will be clear then. We refer to the *shape* of the tableau to be the Young Diagram it is given in. A tableau of shape λ is called a **Standard Tableau** if the entries are the numbers $\{1, ..., m\}$ each occurring only once, where $m = |\lambda|$.

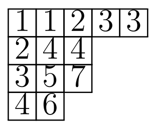
$$T_s = \begin{array}{|c|c|c|c|}\hline 1&3&5&8\\\hline 2&4&9\\\cline{1-3} 6&7\\\cline{1-2}\end{array} \qquad T = \begin{array}{|c|c|c|c|}\hline 1&1&3&4\\\hline 2&3&5\\\cline{1-3} 4&6\\\cline{1-2}\end{array}$$

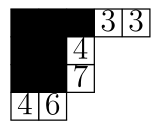
Here, T_s and T are examples of a standard tableau and a tableau on the same shape.

Skew Tableaux generalize this notion for skew diagrams just as one might expect. One could start with two tableaux T_λ and T_ρ of shapes λ and ρ such that $\rho \subset \lambda$. Then the skew tableaux denoted by T_ρ/T_λ has shape $\rho \subset \lambda$ and with filling consistent with that of T_λ. Similarly, one could start with two shapes $\rho \subset \lambda$, produce ρ/λ, and then produce a filling for the skew tableau which is consistent with the two predefined rules. Visually, if $\rho \subset \lambda$ as our last example in the previous subsection, then a skew tableau T_ρ/T_λ of shape $\rho \subset \lambda$ can be given by

$T_\rho/T_\lambda :=$ 

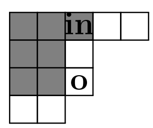
Such a filling would be consistent with the following filling of shape λ:

$T_\lambda :=$ 

$\longrightarrow$ 

It is important to note that the only information of the tableau T_ρ we needed to produce T_ρ/T_λ was its shape ρ. Whatever filling the shape had will still produce the same skew diagram against T_λ.

Lastly, whenever we have a skew diagram or a skew tableau of shape ρ/λ, a box of ρ/λ is called **outside corner** if there are no boxes immediately to the right *and* immediately below that box. Similarly, a box in ρ is called an **inside corner** of ρ/λ if, when ρ is realized by fitting it inside λ, the boxes immediately to the right and immediately below this box of ρ are not contained in ρ. For instance, using our last example, the box designated by the position (3rd row,3rd column) is an outside corner of ρ/λ, whereas as the box (1,3) is an inside corner of ρ/λ. Visually, the box labeled **in** is an inside corner and the one labeled **o** is an outside corner:



2.4 Words on Tableaux

We define a *word* w to be a finite sequence of positive integers. For two words w, w', we define a word operation $w \cdot w'$ through concatenation, and we abbreviate it as ww'. Note that this operation is not commutative in general. Now, let T be a tableau of shape λ, where $l(\lambda) = n$. Then we define $w_i(T)$, or w_i when T is understood, to be the word consisting of the entries of the i^{th} row of T in *increasing* order. Now we define the *word of T* to be

$$w(T) = w_n w_{n-1} \ldots w_2 w_1$$

Note that here T may be a tableau or a skew tableau. Using T_λ, T_ρ/T_λ as in our last example, we have

$$w(T_\lambda) = (46)(357)(244)(11233) \quad \text{and} \quad w(T_\rho/T_\lambda) = (46)(7)(4)(33)$$

where the parenthesis are only used to better visualize the individual row words.

At first sight, is not clear how a given word may be related to a tableau. In fact, not every word can be associated with some young diagram. In order to ensure that a word is associated with a young diagram, the pieces of the word which make up the row words of the diagram must match the specified rule for fillings; that is, the pieces of the word must have weakly increasing lengths and their content must be in weakly increasing order. Now, what can be achieve is a weaker construction, where every word may be associated with a skew tableau, which is not necessarily unique. Let w be a word and partition it in weakly increasing pieces. Call these pieces w_i and think of them as being sub-words of w. Each sub-word can be viewed as a word of a row. Then one obtains a skew diagram by placing each row above and entirely to the right of the preceding piece's row. After describing some important operations one can define on tableaux, we shall come back to this topic and explore the relationship of a tableau and its word in greater depth.

2.5 Row Insertion and Knuth Equivalence

We now introduce the row insertion algorithm on tableaux. This operation will aid us in defining further tableaux operations, which, in turn, will help us in understanding this combinatorial object much better.

The first one we will describe is the simple **row insertion**, or, alternatively, Schensted row insertion. Let T be a tableau and let z be a positive integer. Then, we row insert z into T, abbreviated by the notation $T \leftarrow z$, producing a new tableau T_0 using the following rules:

- If z is as large as all the entries in the first row of T, attach z into a new box at the end of the first row containing z and stop.
- If not, find the leftmost entry of T in the first row such that z is strictly smaller. Let that entry be t_1. Replace t_1 with z.
- Now, compare t_1 with the entries in the second row, and follow the same procedure described in the above bullet points: Either append a new box at the end of the second row (as bullet 1), or *"bump"* the next strictly largest integer compared to t_1; call it t_2.
- Repeat this process inductively until you either find yourself in a situation where you can append the bumped box into the end of a row or, if you have gotten into the last row, and you must bump another box, create a new row containing only the new bumped box.

Example 2.5.1. *We now give two examples for the tableau* $T = \begin{array}{ccccc} 1&1&2&3&3 \\ 2&4&4 \\ 3&5&7 \\ 4&6 \end{array}$. *Firstly,* $T \leftarrow 2$*:*

$$\begin{array}{ccccc} 1&1&2&3&3 \\ 2&4&4 \\ 3&5&7 \\ 4&6 \end{array} \leftarrow 2 \longrightarrow \begin{array}{ccccc} 1&1&2&2&3 \\ 2&2&4 \\ 3&5&7 \\ 4&6 \end{array} \leftarrow 3 \longrightarrow \begin{array}{ccccc} 1&1&2&2&3 \\ 2&2&3 \\ 3&5&7 \\ 4&6 \end{array} \leftarrow 4 \longrightarrow \begin{array}{ccccc} 1&1&2&2&3 \\ 2&2&3 \\ 3&4&7 \\ 4&6 \end{array} \leftarrow 5 \longrightarrow \begin{array}{ccccc} 1&1&2&2&3 \\ 2&2&3 \\ 3&4&7 \\ 4&5 \\ 6 \end{array}$$

Further, if $(T \leftarrow 2) \leftarrow 3$*, then*

$$\begin{array}{ccccc} 1&1&2&2&3 \\ 2&2&3 \\ 3&5&7 \\ 4&5 \\ 6 \end{array} \leftarrow 3 \longrightarrow \begin{array}{cccccc} 1&1&2&2&3&3 \\ 2&2&4 \\ 3&5&7 \\ 4&6 \end{array} = (T \leftarrow 2) \leftarrow 3$$

We note that this algorithm always produces a new, well defined, tableau. To see this, first note that the new tableau remains weakly increasing in each row. Also, note that an entry x bumps a box containing an entry y in the a given row in a tableau if and only if $y > x$ and, if z is another entry in that row such that z is to the left of y, then $z \leq x$. Thus, the bumped entry y can only move down and to the left in comparison to the first column it was first sitting in the tableau. But, our original tableau is strictly increasing in every column so that the entry directly above the new position of y is at most x, which is strictly less than y by assumption.

The latter shows that the row insertion algorithm is reversible, provided that we are told which box was last added. For instance, one might start with a Tableau T and a box with entry x which was last added by the row insertion procedure. Then, one wishes to recover T_0 and t such that $T_0 \leftarrow t = T$. If the last-added box is in the first row, then it is necessarily the rightmost box in that row. In this case, t is equal to the entry in the rightmost box, *i.e* $t = x$. If the box is not in the first row, then one simply finds the rightmost entry in the row above which is strictly less than x. Repeat this procedure inductively, recording the entries, until arriving in the first row, where the entry bumped out of the first row is the desired t. We highlight this procedure using a gray path on our previously defined T and assuming that the box with entry 6 (in the last row) is the one last added by the algorithm

1	1	2	3	3
2	4	4		
3	5	7		
4	6			

In this example, $T = (T_0 \leftarrow 3)$ where $T_0 = \{11234, 245, 367, 4\}$

Moreover, this grey path denoted above is called the *bumping route* of 3 when inserted in T_0. We commonly use the letter R to denote a bumping route. As one might expected, such object is of key importance in the study of row insertion. The bottom-most box of the bumping route is referred to as the *new box* of the row-insertion. In this last example, the box in position (4,2) would be the new box of the row-insertion $T = (T_0 \leftarrow 3)$. We now give some brief definitions of concepts related to routes.

We call a route R **weakly left** (respectively strictly) of another bumping route $\bar{R}$, if in each row containing a box of $\bar{R}$, the route R has a box in the same row which is weakly (respectively strictly) left of that of $\bar{R}$. We are now ready for our first useful proposition related to the row bumping algorithm:

Proposition 2.5.2. *Let T_0, with shape ρ, and $U := (\cdots((T_0 \leftarrow t_1) \leftarrow t_2) \leftarrow \cdots \leftarrow t_{k-1}) \leftarrow t_k$, with shape λ. If $\{t_1, t_2, \cdots, t_k\}$ is a sequence of weakly increasing positive integers, then no two boxes in λ/ρ lie in the same column.*

Conversely, suppose U is a tableau on $\lambda \supset \rho$ such that it has exactly k boxes on λ/ρ lying in different columns. Then, there is a unique tableau T_0 and a unique set of of weakly increasing positive integers $\{t_1, t_2, \cdots, t_k\}$ such that $U = (\cdots((T_0 \leftarrow t_1) \leftarrow t_2) \leftarrow \cdots \leftarrow t_{k-1}) \leftarrow t_k$

To prove this proposition, it will be of our interest to prove a result famously referred to as *row bumping lemma*

Lemma 2.5.3 (Row Bumping lemma)**.** *Let T be a tableau. Let R denote the bumping route of row inserting x into T and let R' denote be the bumping route of row inserting x' into $T \leftarrow x$. Let b and b' denote the new boxes produced by the row insertions related R and R' respectively. Then,*

Case 1. *If $x > x'$, then R' is weakly left of R and b' is weakly left and strictly below b*

Case 2. *If $x \leq x'$, the R is strictly left of R', and b is strictly left and weakly below b'*

Proof. We first consider the first case, where $x > x'$. If x is placed at the end of the first row, then the result follows trivially. Now, assume otherwise and let y and y' denote the elements bumped by x and x' respectively. Note that x' bumps the least entry strictly grater than it but also $x > x'$; hence, y' is at most x, *i.e* $y' \leq x$. But since row insertion produces a well defined tableau, this means that the box of y' cannot be to the right of the box of x. Noting that $y' \leq x$ and $x < y$, we apply this argument inductively, for each row of the tableau. This shows that R' is weakly left of R. Furthermore, since $y' < y$, we will never place the new box of the route R', *i.e* b, at the end of a row where the route R is present. Hence, b' is weakly to the left and strictly below b as required.

For the second case, assume that x or x' are placed at the end of the first row, then the result follows trivially. Now, assume otherwise and let y and y' denote the elements bumped by x and x' respectively. Consider the first row of $T \leftarrow x$. Note that y' must lie strictly to the right of the box where x is in $T \leftarrow x$, because all the boxes weakly to the left of the box where x occupied in $T \leftarrow x$ are filled by elements that are weakly smaller than x, and hence, weakly less than or equal to x'. Thus, this implies that y is strictly left of y' in T, in particular, $y \leq y'$. Again, we apply this argument inductively to each subsequent row to conclude that R is strictly left of R'. Because of this, R cannot stop above any row that R' stops. Hence, if R and R' stop at different rows, it must be that b is weakly below b' and hence, since R is strictly left of R' so is b in relation to b'. Finally, if they stop at the same row, the same reasoning shows that b is strictly left of b' as required. ■

With the *row bumping lemma* in our toolbox, proposition 2.5.2 becomes an easy extension of it.

Proof of 2.5.2. We apply the row bumping lemma inductively, denoting by R_i the route related to inserting t_i into the tableau $[(\cdots((T_0 \leftarrow t_1) \leftarrow t_2) \leftarrow \cdots \leftarrow t_{i-1})]$ and b_i denoted the new box of the row insertion associated with R_i. Now, note that the boxes of λ/ρ are precisely the new boxes of the subsequent row insertions of the t_i. But, by the row bumping lemma, R_i is strictly to the left of R_{i+1} and b_i is strictly left and weakly below b_{i+1}, for all i in $[k-1]$. Hence, no two boxes of λ/ρ lie in the same column as required.

For the converse, simply note that, as argued before, the row insertion algorithm is reversible. Start by performing the reverse row bumping algorithm on U from the box located at the right-most of λ/ρ. This will give you a tableau T_k and an integer t_k such that $T_k \leftarrow t_k = U$. Next, perform the reverse row bumping algorithm on the resulting tableau T_k from the box located secondly right-most of λ/ρ. This will give you a tableau T_{k-1} and an integer t_{k-1} such that $[(T_{k-1} \leftarrow t_{k-1}) \leftarrow t_k] = U$. Also, since λ/ρ is a well defined skew tableau, it follows that $t_{k-1} \leq t_k$. Inductively, we get the desired result. ■

An analogous proposition of that of 2.5.2 can be formulated to explain the behavior of the row insertion process when dealing with strictly increasing sequences.

Proposition 2.5.4. *Let T_0, with shape ρ, and $U := (\cdots((T_0 \leftarrow t_1) \leftarrow t_2) \leftarrow \cdots \leftarrow t_{k-1}) \leftarrow t_k$, with shape λ. If $\{t_1, t_2, \cdots, t_k\}$ is a sequence of strictly decreasing positive integers, then no two boxes in λ/ρ lie in the same row.*

Conversely, suppose U is a tableau on $\lambda \supset \rho$ such that it has exactly k boxes on λ/ρ lying in different rows. Then, there is a unique tableau T_0 and a unique set of of strictly decreasing positive integers $\{t_1, t_2, \cdots, t_k\}$ such that $U = (\cdots((T_0 \leftarrow t_1) \leftarrow t_2) \leftarrow \cdots \leftarrow t_{k-1}) \leftarrow t_k$

Proof. The proof is analogous to that of 2.5.3, crucially using the first case of the row bumping lemma. The only meaningful difference to note here is that, for the converse direction, the reverse bumping algorithm should be applied from the bottom-most to the uppermost row of λ/ρ. ■

The above presented propositions showcase how the row insertion algorithm can be understood through diagrammatically lens on the tableaux. However, it is often useful to understand how this algorithm affects words of tableaux. For simplicity, let's first consider that an entry x is being inserted in some row of a Tableau T. Consider that row's word specifically; denote it by w. Remember that w is a weakly increasing sequence of positive integers. Write $w = u \cdot v$, where u and v are words such that for any integer in u_i in u, we have that $u_i \leq x$, and, for any integer in v_i in v, $v_i > x$. Here, it is possible that either v or u are empty words. The row insertion algorithm tells us to find y in this row such that y is the left-most entry which is strictly greater than x, bumping y down. Thus, we can further rewrite this word as $w = u \cdot y \cdot v$. If such y exists, the row insertion is performed by precisely replacing this y with x, bumping the former into the next row. In this case, one can visualize the change performed by the row insertion algorithm in the word of this row as follows:

$$w \cdot x = (u \cdot y \cdot v) \cdot x \rightsquigarrow y \cdot (u \cdot x \cdot v)$$

If no such y exists, we simply append x to the end of the row word w. This process becomes more interesting when we allow for the entire word of the tableau to be considered. This is best understood through an example. For this, we will continue to use the previously defined T this subsection. Here, $w(T) = (46)(357)(244)(11233)$. Now, row inserting 2 into T changes its word as follows:

$$\begin{aligned}(46)(357)(244)(112 \cdot 3 \cdot 3) \cdot 2 &\rightsquigarrow (46)(357)(244) \cdot 3 \cdot (11223)\\ &\rightsquigarrow (46)(357) \cdot 4 \cdot (234)(11223)\\ &\rightsquigarrow (46) \cdot 5 \cdot (347)(234)(11223)\\ &\rightsquigarrow (6)(45)(347)(234)(11223)\end{aligned}$$

This relationship between row insertion and words was first presented by Donald Knuth, who years later rediscovered the row insertion algorithm first presented by Schensted. The way the algorithm acts on words is of extreme importance in proving further results that will be presented later. For the moment, we note that this relationship reveals the inner structure of the algorithm, breaking it down into atomic pieces. To see this, let T be a tableau w be its corresponding first row word. Pretend that we are to row insert x into T. Again, we choose a factorization of w such that y represents the element of the first row to be bumped by the insertion algorithm. Then, $w = u \cdot y \cdot v$ of w. Let $v = v_1 \cdots v_q$. We declare the relationship between two words $k < p$ whenever every integer in k is smaller than or equal to every integer in p.

The steps can be listed, with the rules that govern them, as

$$\begin{aligned}(uyv_1 \cdots v_{q-1}v_q) \cdot x = uyv_1 \cdots v_{q-1}v_q x &\longmapsto uyv_1 \cdots v_{q-1}xv_q && (x < v_{q-1} \leq v_q)\\ &\longmapsto uyv_1 \cdots v_{q-2}xv_{q-1}v_q && (x < v_{q-2} \leq v_{q-1})\\ \cdots &\longmapsto uyv_1xv_2 \cdots v_{q-1}v_q && (x < v_1 \leq v_2)\\ &\longmapsto uyxv_1 \cdots v_q && (u < y < x \leq v_1)\end{aligned}$$

Note that the reiterated transformation at each step is:

$$xyz \longmapsto xzy \qquad (z < x \leq y) \tag{K'}$$

At this point, the row insertion algorithm tells us that y is bumped into the next row, while x stays at that given position. Thus, we have to completely bump y out of this row. Again, we list the steps as follow

$$\begin{aligned} u_1u_2\cdots u_{p-1}u_pyxv &\longmapsto u_1u_2\cdots u_{p-1}yu_pxv && (u_p \leq x < y) \\ &\longmapsto u_1u_2\cdots u_{p-2}yu_{p-1}u_pxv && (u_{p-1} \leq u_p < y) \\ \cdots &\longmapsto u_1yu_2\cdots u_pxv && (u_2 \leq u_3 < y) \\ &\longmapsto yu_1u_2\cdots u_pxv && (u_1 \leq u_2 < y) \end{aligned}$$

Through these reiterated transformations, the rule governing them is:

$$xyz \longmapsto yxz \qquad (x \leq z < y) \tag{K''}$$

It turns our that these two word transformations, K'' and K', are so important that they take a specific name: **elementary Knuth transformations**.

We call two words w and w', **Knuth equivalent** if one is obtained from another by solemnly applying Knuth transformations to them. This defines an equivalence relation on the set of words of tableaux. The latter will be denoted by $w \equiv w'$.

In particular, these two operations motivate a proposition, whose proof is immediate from this discussion.

Proposition 2.5.5. *Let T be any tableau and t be any positive integer, then $w(T \leftarrow t) \equiv w(T) \cdot t$*

At this point, it is hard to see how Knuth equivalence will aid us any further than describing an algorithm to go from a word of tableau to a new word. As an example, this notion is crucial in proving the following important theorem.

Theorem 2.5.6. *Every word w is Knuth equivalent to some word of a* ***unique*** *tableau T.*

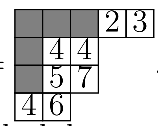
Proof. Let $w = x_1x_2\cdots x_k$. Then, let $T_1 = \boxed{x_1}$. Then, T can be obtained by setting $T := (\cdots((T_1 \leftarrow x_2) \leftarrow x_3)\cdots \leftarrow x_{k-1}) \leftarrow x_k$. By our last prop $w(T) \equiv x_1 \cdot x_k = w$. Unfortunately, uniqueness is not so easy to show, but, in terms of future applications, even more important than existence. Since the purpose of this paper is to elucidate the applications of the Combinatorics of young tableaux, we omit the proof of uniqueness from this discussion. However, the interested reader may find a proof of this result in page 22 of Fulton's [1]. ∎

2.6 Jeu de Taquín

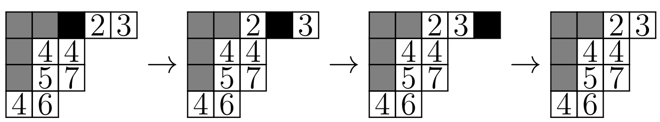
We now introduce another algorithm on tableaux. This algorithm is commonly named **sliding game** or **Jeu de Taquín**, which literally means *teasing game* when translated from French. From Fulton's [1]: "The name *jeu de taquin* refers to the French version of the 15 puzzle, in which one tries to rearrange the numbers by sliding neighboring squares into the empty box." Indeed, the chosen allusion makes sense, as we will see in a moment.

In contrast with the row-insertion algorithm described last section, this algorithm acts on skew tableaux in such way to turn them into well defined tableaux. Given a skew tableau S, the following is what we call the *sliding algorithm*:

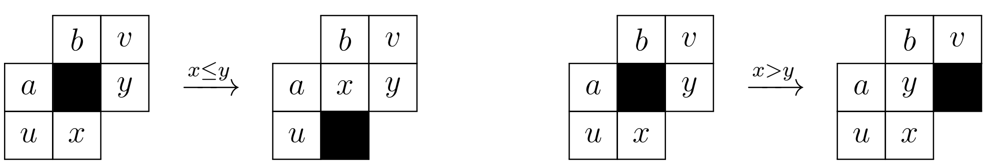
- Choose an inside corner (as defined in 2.3) of S. (As a reminder, an inside corner of skew tableau will be a box with no filling)
- Consider the two neighboring entries of such an inside corner. These are the entries immediately below and immediately above that empty box. These are guaranteed to exist, since we are starting with an outside corner. Identify which entry is smaller. If they happen to have the same size, simply choose the one directly below the inside corner (*i.e* , the leftmost one).
- *Slide* the smaller of the two into the inside corner. Here, *slide* literally means to exchange the inside corner with the smaller outside corner.
- Repeat the previous two steps with the empty box, identifying its two new neighboring entries after the sliding procedure. Repeat, inductively until an inside corner becomes an outside corner.
- At this point, remove the inside corner from the diagram. Halt.

Again, this is best explained visually. Take the skew tableau $S =$ 

. Then, the sliding procedure, choosing the inside corner conveniently depicted by the black box, produces:

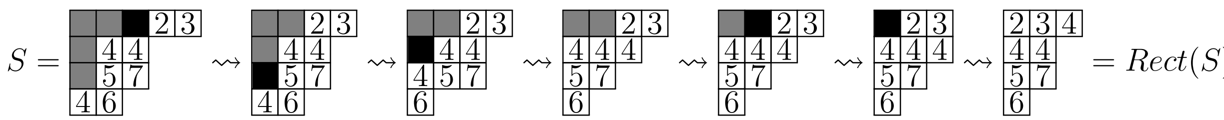


We will address the issue of choosing an inside corner in what follows, but first, it is important to note that this procedure always produces a well defined skew tableau. The fact that it produces a skew diagram is evident, but in order to verify that the fillings remain consistent, one must check two cases. Consider the following sliding results:



In the first case, all the rows remain weakly increasing since $x \le y$ and the column fillings are not affect. In the second case, the row fillings are not affected, but since $x > y$ the columns remain strictly increasing. This shows that the sliding procedure always produce a well defined skew tableau. Furthermore, as with the row-insertion algorithm, the sliding algorithm is again reversible, provided that we know what outside corner was removed last.

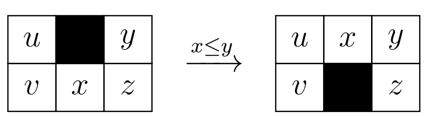
Now, after choosing *some* a inside corner of S, and applying the sliding algorithm, one is faced with the option to "tease" the resulting skew tableau again by choosing another inside corner (if there is any). Indeed, one can do this inductively, choosing inside corners and sliding them out, until one is left with a tableau. This resulting tableau is called called the **rectification of S**, denoted in short by $Rect(S)$, and the entire procedure of perturbing S to arrive at $Rect(S)$ is what *jeu de taquín* is. To give an example, consider S as above. Then, using the black boxes to indicate our choice of inside corners to run the sliding algorithm at each step, jeu de taquin gives:



One may be justifiably concerned about the well-definiteness of this procedure, since it involves a *choice* of inside corners repeatedly. However, we will know show that, in fact, this procedure is independent of the sequence of choices of inside corners. That is, no matter how one chooses the inside corners of S to apply the sliding algorithm, $Rect(S)$ will always be the same. Firstly, a proposition:

Proposition 2.6.1. *If a skew tableau is obtained by another through a sequence of slides, then their words are Knuth equivalent.*

Proof. Firstly, it is important to note that a sequence of slides may not produce a skew tableau. Unless, you have fully thrown away the inside corner you have started with, what you have is a skew tableau with a puncture box in it. However, it is still possible to define a word for this object under these circumstances; such a word is defined in the most obvious way, ignoring (or skipping) the punctured box in this object. Thus, it is easy to see that if a horizontal slide is performed, then the word of the tableau remains the same. Thus, if the slides are horizontal, the words of the tableau are the same, and the proposition follows easily. Now, consider a vertical slide in the following example, where $u < v \leq x \leq y < z$:



Here, the word changes from $vxzuy$ into $vzuxy$. Indeed, this is achieved by the following Knuth transformations:

$$\begin{aligned} vxzuy &\equiv vxuzy && (K') \\ &\equiv vuxzy && (K') \\ &\equiv vuzxy && (K'') \\ &\equiv vzuxy && (K'') \end{aligned}$$

Now things get a bit more technical when we allow for more than four corners as depicted above. This case is handled by a tedious induction argument and picky notation. We omit the latter from this present discussion. For the interested reader, a proof of this case is presented in page 21 of Fulton's [1]. ■

We have now set ourselves up to success in order to prove what we originally wanted:

Claim 2.6.2. *Jeu de taquin procedure is well defined: Given a skew tableau S, $Rect(S)$ is independent of the order that one chooses to eliminate inside corners.*

Proof. This claim follows from last section's **theorem** 2.5.6 and **proposition** 2.6.1 shown above. ■

This last claim can be rephrased to give other useful characterization of $Rect(S)$. We will label these as corollaries, since their proofs, just like the one presented in the above claim, follow directly from 2.5.6 and 2.6.1.

Corollary 2.6.3. *Let S be a skew tableau and let $w(S)$ represent its word. Then, $Rect(S)$ is the* ***unique*** *tableau T such that $w(T) \equiv w(S)$.*

Proof. As above. ■

Corollary 2.6.4. *Let S and S' be skew tableaux. Then,*

$$Rect(S) = Rect(S') \iff w(S) \equiv w(S')$$

Proof. As above. ■

2.7 Product of Tableaux

We now come to the last introductory operation on tableaux that we will present in this report. This operation is most commonly thought as a *multiplication* of tableaux because of its suggestive notation. There are a couple of ways that one may choose to define this operation. Since the algorithms and operations were recently introduced in this discussion, we will first define the product in terms of those.

Let T and U be tableaux, and denote by $w(T)$ and $w(U)$ their words. Let A be the unique tableau whose word is Knuth equivalent to $w(T) \cdot w(U)$ (concatenation of words, as defined before). Then, we define the product of T with U by

$$A =: T \cdot U$$

Since A is unique by 2.5.6, this operation is well defined. Interestingly, this operation can be defined naively, but in a analogous fashion, using yet another operation on tableaux.

For two tableaux T and U, define the skew tableau $T * U$ by taking a rectangle of empty squares with dimensions *#rows of* $T \times$ *#columns of* U and gluing T directly below this rectangle and U directly right and of it.

As an example, let $T = \begin{array}{|c|c|c|c|}\hline a & b & c & d \\ \hline e & f \\ \cline{1-2}\end{array}$ and $U = \begin{array}{|c|c|c|}\hline u & v & w \\ \hline x \\ \cline{1-1} y \\ \cline{1-1}\end{array}$ .Then,

$$T * U = \begin{array}{|c|c|c|c|c|c|c|}\hline \blacksquare & \blacksquare & \blacksquare & \blacksquare & u & v & w \\ \hline \blacksquare & \blacksquare & \blacksquare & \blacksquare & x \\ \cline{1-5} \blacksquare & \blacksquare & \blacksquare & \blacksquare & y \\ \cline{1-5} a & b & c & d \\ \cline{1-4} e & f \\ \cline{1-2}\end{array}$$

Then, one defines $T \cdot U := \mathit{Rect}(T * U)$. Indeed, since $w(T * U) = w(T) \cdot w(U)$ and jeu de taquin guarantees the uniqueness of $\mathit{Rect}(T * U)$, 2.6.3 guarantees that the above two different definitions of the product are equivalent.

Proposition 2.7.1. *Let $\mathcal{T}$ denote the set of tableaux. Then, $(\mathcal{T}, \cdot)$ is a monoid with identity element denoted by $\emptyset$, which is the empty tableau associated with the empty partition $\emptyset$.*

Proof. Clearly, for any $T \in \mathcal{T}$, we have $T \cdot \emptyset = \emptyset \cdot T = T$, so that $\emptyset$ is indeed the identity element under this operation. It remains to show that $\cdot$ is associative.

Let $T, U, V \in \mathcal{T}$. Consider, $(T \cdot U) \cdot V = \mathit{Rect}(T * U) \cdot V$. Similarly $T \cdot (U \cdot V) = T \cdot \mathit{Rect}(U * V)$. But, note that $\mathit{Rect}(T * U) \cdot V$ and $T \cdot \mathit{Rect}(U * V)$ are the unique tableaux whose words is Knuth equivalent to $w(\mathit{Rect}(T * U)) \cdot w(V)$ and $w(T) \cdot \mathit{Rect}(U * V)$, respectively. But, since concatenation of words are associative, we have:

$$w(\mathit{Rect}(T * U)) \cdot w(V) \equiv (w(T) \cdot w(U)) \cdot w(V) = w(T) \cdot (w(U) \cdot w(V)) \equiv w(T) \cdot \mathit{Rect}(U * V)$$

Hence, $(T \cdot U) \cdot V = T \cdot (U \cdot V)$ as required. ■

There is yet another construction of the product which is worth mentioning in this report because it ties together the row-insertion algorithm and jeu de taquin. As before, let $T, U \in \mathcal{T}$, and let $w(T) = t_1 t_2 \cdots t_n$ and $w(U) = u_1 u_2 \cdots u_k$ be their words. Then,

$$T \cdot U := (\cdots((T \leftarrow u_1) \leftarrow u_2) \cdots \leftarrow u_{k-1}) \leftarrow u_k$$

As a consistency check, applying 2.5.5 inductively, we have that $w(T \cdot U) \equiv w(T) \cdot u_1 u_2 \cdots u_n = w(T) \cdot w(U)$. Hence, this definition agrees with the previous ones.

Note that, though $\cdot$ is associative, it is not a commutative in general. The latter is easily shown using this last definition of the product:

$$\begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline \end{array} \cdot \begin{array}{|c|}\hline 2 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline 1 & 2 & 2 \\ \hline 3 \\ \cline{1-1} \end{array} \neq \begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline 2 \\ \cline{1-1} \end{array} = \begin{array}{|c|}\hline 2 \\ \hline \end{array} \cdot \begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline \end{array}$$

3 Applications to Counting Problems

This section is dedicated to results related to counting problems that are solved using the theory developed thus far. Obviously, the theorems and ideas presented in this sections are not exhaustive and one may want to refer to [1], [3], [2], for more results concerning counting problems.

3.1 RSK Correspondence

The acronym *RSK* stands for Robinson-Schensted-Knuth. This result is quite marvelous on its own. *RSK* correspondence, due to Donald Knuth, is in fact a generalization of a weaker correspondence named **Robinson-Schensted** correspondence. Here is what it states:

Theorem 3.1.1 (Robinson-Schensted Correspondence)**.** *There is a bijective correspondence between words of length r on the alphabet of $[n]$ and pairs of semi-standard tableaux (P, Q), where P is a semi-standard tableau with entries in $[n]$ and with associated partition $\lambda \vdash r$ and Q is a standard tableau on $[r]$ on the same shape λ.*

We have already seen in 2.5 how a word and a tableau may be related to one another. More specifically, we have shown that every word is *Knuth equivalent* to a word of a unique tableau. In other words, if one starts with a word $u = u_1 u_2 \cdots u_r$ and let

$$P(u) := (\cdots((\boxed{u_1} \leftarrow u_2) \leftarrow u_3) \cdots \leftarrow u_{r-1}) \leftarrow u_r$$

then, $P(u)$ does not correspond uniquely to u. In fact, choose any other word $u' \equiv u$. Then, $P(u) = P(u')$. This is precisely the reason for including pairs of tableaux in this correspondence.

If we have any hopes to achieve a correspondence between words and tableaux, we must take into account Knuth transformations. This is achieved by recording where each entry of the word u is placed throughout the standard procedure of row insertion. The way we record these placements is through what is called *recording tableau.* The latter is commonly denoted by the letter Q in the correspondence, while the tableau obtained through the standard procedure of row insertion is called the *insertion tableau*, denoted by P. As before, we use the notation $P(u)$ and $Q(u)$ to denote the insertion and recording tableaux respectively produced by the word u. A formal definition of Q is given as follows:

Definition 3.1.2. *Let $u = u_1 u_2 \cdots u_r$. For all $k \leq r$, denote by $P_k(u)$ the tableau obtained by row inserting $\{u_1, \cdots, u_k\}$ by the standard procedure. Similarly, let $Q_1(u) = \boxed{1}$, and, inductively for $k > 1$, let $Q_k(u)$ be the tableaux $Q_{k-1}(u)$ with an additional box filled with content k, placed at the position where u_k was placed in $P_k(u)$. We define $Q(u) := Q_r(u)$.*

One thing to note from this definition is that the recording tableau Q will have the same shape as the insertion tableau P. In particular, since P is always a well defined young tableau, Q is indeed a young diagram. But, since the box added to P_k through the standard procedure is always an outside corner, so is the box added to Q_k, with entry always largest than any other already in Q_{k-1}. More importantly, with entry larger than any entry above or to the left of it. Hence, Q is indeed a standard tableau, since its entries are $\{1, \cdots, r\}$ each with multiplicity one.

We now give an example of this procedure with the word $u = 1336572$:

$$(P_1 = \begin{array}{|c|}\hline 1 \\ \hline\end{array}) \leftarrow 3 \quad \longrightarrow (P_2 = \begin{array}{|c|c|}\hline 1 & 3 \\ \hline\end{array}) \leftarrow 3 \quad \longrightarrow (P_3 = \begin{array}{|c|c|c|}\hline 1 & 3 & 3 \\ \hline\end{array}) \leftarrow 6 \quad \longrightarrow (P_4 = \begin{array}{|c|c|c|c|}\hline 1 & 3 & 3 & 6 \\ \hline\end{array}) \leftarrow 5$$

$$\longrightarrow (P_5 = \begin{array}{|c|c|c|c|}\hline 1 & 3 & 3 & 5 \\ \hline 6 \\ \cline{1-1}\end{array}) \leftarrow 7 \quad \longrightarrow (P_6 = \begin{array}{|c|c|c|c|c|}\hline 1 & 3 & 3 & 5 & 7 \\ \hline 6 \\ \cline{1-1}\end{array}) \leftarrow 2 \quad \longrightarrow P_7 = \begin{array}{|c|c|c|c|c|}\hline 1 & 2 & 3 & 5 & 7 \\ \hline 3 \\ \cline{1-1} 6 \\ \cline{1-1}\end{array} = \mathbf{P(u)}$$

Now, we use the insertion route of $P(u)$ to compute $Q(u)$:

$$Q_1 = \begin{array}{|c|}\hline 1 \\ \hline\end{array} \quad \longrightarrow Q_2 = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline\end{array} \quad \longrightarrow Q_3 = \begin{array}{|c|c|c|}\hline 1 & 2 & 3 \\ \hline\end{array} \quad \longrightarrow Q_4 = \begin{array}{|c|c|c|c|}\hline 1 & 2 & 3 & 4 \\ \hline\end{array}$$

$$\longrightarrow Q_5 = \begin{array}{|c|c|c|c|}\hline 1 & 2 & 3 & 4 \\ \hline 5 \\ \cline{1-1}\end{array} \quad \longrightarrow Q_6 = \begin{array}{|c|c|c|c|c|}\hline 1 & 2 & 3 & 4 & 6 \\ \hline 5 \\ \cline{1-1}\end{array} \quad \longrightarrow Q_7 = \begin{array}{|c|c|c|c|c|}\hline 1 & 2 & 3 & 4 & 6 \\ \hline 5 \\ \cline{1-1} 7 \\ \cline{1-1}\end{array} = \mathbf{Q(u)}$$

If a bijection exists, there must be a way to go from (P, Q), where the pairs have the desired properties, to some word u with length r. This is exactly where Q comes in handy. Remember that Schensted algorithm is completely reversible provided that we know which box was last added. That is, there is no ambiguity to what entry was last added provided we now which box the row insertion algorithm created. Indeed, Q is created to record the order in which the boxes in P were added. Using Q, we are able to completely reverse P.

This method can be more accurately described (backwards) inductively as follows. Start with $(P_r, Q_r) = (P, Q)$ and located the box in Q occupied by the entry r. Locate the corresponding box in P. Then, proceed by running backwards row-insertion algorithm described in 2.5. Because of the construction of P, the result of this first step will be a tableau P_{r-1} and an entry x such that $(P_{r-1} \leftarrow x) = P_r = P$. Let $x = u_r$. To produce Q_{r-1} simply remove the box containing r from Q. At each step k, locate the entry k in Q_k and run this same algorithm, denoting by u_k the entry such that $(P_{k-1} \leftarrow u_k) = P_k$. Note that at each inductive step, the largest entry in Q_k will always be an outside corner, so the reverse process in indeed well defined. By the reversibility of Schensted algorithm under the special condition that we know which box was last created, we are able to uniquely recover this word $u = u_1 \cdots u_r$.

Hence, this discussion has shown that a word u of length r in the alphabet of $[n]$ completely determines a pair (P, Q) with the desired properties described in the enunciation of 3.1.1. Thus, the correspondence is shown.

In the above correspondence, we are stipulating that Q is a standard tableau. What if P and Q have yet the same shape, but nothing is *expected* of Q? That is, what if Q is just a tableau with entries in $[m]$? Indeed, this extended case is due to Knuth himself and is what is called the *RSK* correspondence. We will first introduce some notation in order to formalize this result.

We call $w = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$, where $u_1 \cdots u_r$ and $v_1 \cdots v_r$ are words of length r, a *two-rowed array of length r*. In particular, we can interpret $w = \left(\begin{smallmatrix} 1 & 2 & \cdots & r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$ as the word $u_1 \cdots u_r$. This notation allows us to record the above reverse process by a sequence of two rowed arrays $w_k = \left(\begin{smallmatrix} k & k+1 & \cdots & r \\ u_k & u_{k+1} & \cdots & u_r \end{smallmatrix}\right)$, where u_k is such that $(P_{k-1} \leftarrow u_k) = P_k$. Using this notation, we describe the algorithm which will lead us to Knuth's generalization of this correspondence. The procedure requires only minor modification from the above process.

Starting from a pair (P, Q) of tableaux on the same shape $\lambda \vdash r$, where P has entries on $[m]$ and Q in $[n]$. Start by locating the box in $Q = Q_r$ with the largest entry. If there are more than one box with largest entries, choose the farthest to the right (they are not in the same column). Call that entry v_r and remove it from Q_r to get Q_{r-1}. Locate the corresponding box in $P_r = P$ and perform backwards row insertion in that box. Just as before, you will end up with an entry x and a tableau P_{r-1} such that $(P_{r-1} \leftarrow x) = P$. Again, let $u_r := x$. Define $w_r := \left(\begin{smallmatrix} v_r \\ u_r \end{smallmatrix}\right)$. Inductively, just as before, we get a two-rowed array $w := w_1 = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$

Note that, by construction, $v_1 \leq v_2 \leq \cdots \leq v_r$ (1). Also, by 2.5.3 the bottom row of the two-word array will satisfy $u_{k-1} \leq u_k$ whenever $v_{k-1} = v_k$ (2). If a two-rowed array is arranged in this particular way (when (1) and (2) hold), we say that w is in **lexicographic order**, or that w is a **lexicographic array**. Another way to describe this is to say the pair $\begin{pmatrix} v \\ u \end{pmatrix}$ is placed to the right of the pair $\begin{pmatrix} v' \\ u' \end{pmatrix}$ if and only if $(v' < v)$ or if $(v' = v$ and $u' \leq u)$. Thus, starting from (P, Q) as above, we end up with a two-rowed array w arranged in lexicographic order.

Now, given a two-rowed array in lexicographic order $w = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$, we construct $P(w)$ by the canonical procedure of row insertion using the bottom row of w, but now, the entries of $Q(w)$ are given by the respective entries in the top row of w. Note that in order for this process to generate a well defined pair of tableaux (P, Q) with the desired properties, the condition that w is in lexicographic order is necessary and sufficient. $P(w)$ will always be a tableaux, but the construction of $Q(w)$ is in jeopardy if we do not impose this condition. In particular, if $v_i = v_j$ for $i \leq j$ then one needs that $u_i \leq u_j$ so that 2.5.3 guarantees that the box added to Q_i is strictly left and weakly below the box added to Q_j, and hence, Q_j is a tableau and so is $Q_r = Q$.

Now, a couple of observations are due before we finally state the RSK correspondence theorem. We call a two-rowed array w a **word** when we can view one as such, *i.e* when $w = \left(\begin{smallmatrix} 1 & 2 & \cdots & r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$. Furthermore, if the word in the bottom row has entries in $[r]$ each with multiplicity one, we call w a **permutation**. This is not unmotivated since, with this notation, we can think of w as an element $\sigma \in S_r$.

Theorem 3.1.3 (Robinson-Schensted-Knuth Correspondence)**.** *There is a bijective correspondence between two-rowed lexicographic arrays w and ordered pairs (P, Q) of tableaux on the same shape λ. In particular, if we fix the length of the word $w = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$ to be r, then there is a bijective correspondence between these words w and pairs of tableaux (P, Q) on the same shape λ, where $\lambda \vdash r$.*

Proof. The proof follows immediately from the above prescribed algorithm for arbitrary lexicographic two-rowed arrays. ■

We immediately see that 3.1.1 is a special case of the above theorem for the case that the two-rowed lexicographic array w is taken to be a word, since every two-rowed array which is a word is in lexicographic order. Also, a useful corollary which is again a special case of the above theorem for when w is a permutation states the following:

Corollary 3.1.4. *There is a bijective correspondence between permutations w and pairs of* ***standard tableaux*** *(P, Q) on the same shape. In particular, if w is taken to be a permutation on r letters, then, there is a bijective correspondence between such w and pairs of* ***standard tableaux*** *(P, Q) on the same shape λ, where $\lambda \vdash r$.*

Furthermore, we can generalize 3.1.3 even further. Namely, we can view this theorem as a correspondence between matrices A with non-negative integer entries and pairs of tableaux (P, Q) with the same shape. To do so, a few cleaver observations are in order.

Firstly, note that any two-rowed array $w = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$ gives rise to a *unique* two-rowed array in lexicographic order by a sequence of permutation of the columns of w. In turn, there is a natural correspondence between two-rowed arrays in w and matrices A with non-negative entries. Such is given by the following: Start with a two rowed array w, whose top and bottom row consists of entries in $[m]$ and $[n]$ respectively. We can think of w as a collection of columns of the form $\binom{i}{j}$, where $i \in [m]$ and $j \in [n]$. Then, we construct $A(w)$, an $m \times n$ matrix, by letting its (i, j) entry be the number of times the column $\binom{i}{j}$ appears in w. Conversely, starting from a $m \times n$ matrix A with non-negative integer entries, we can arrive in a two-rowed array w by letting such be the unique (up to lexicographic rearrangement) two-rowed array composed of columns $\binom{i}{j}$, each appearing with multiplicity $a_{i,j}$.

Through this construction, we see that the *RSK correspondence* 3.1.3 extends to a correspondence between matrices A with non-negative integer entries and pairs of tableaux (P, Q) with the same shape. In particular, if we fix matrices A of dimension $m \times n$, then the above algorithm tells us that these are in bijection with pairs of tableaux (P, Q), where Q has entries in $[m]$ and P has entries in $[n]$.

Just as before, we may wonder under what conditions does the matrix A correspond to a word or a permutation w. Note that the i^{th} row sum of A is precisely the number of entries i in Q. Likewise, the j^{th} row sum of A is the number of entries j in P. Hence, A corresponds to a word w if and only if it's associated Q is a standard tableau, *i.e* if and only if each row of A consists only of a single entry 1, and all other entries 0. Likewise, A corresponds to a permutation if and only if the latter holds for both rows and columns. If this holds, A can also be taught as a permutation matrix itself.

This construction is useful in many ways. Remarkably, this idea of associating (P, Q) with a matrix A can be useful in providing a proof of 3.1.3 that does not rely on row insertion at all! The latter is uses an algorithm known as *the matrix-ball construction*. We will not provide an explicit construction of this algorithm, but the interest reader may want to refer to Section 4.2, in Fulton's book.

In turn, this algorithm produces an immediate proof of the following theorem, whose corollaries will be useful later in this report.

Theorem 3.1.5 (Symmetry Theorem). *If $w = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$ is a two rowed array corresponding to the pair of tableaux (P, Q), then (Q, P) corresponds to the two-rowed array $w^{-1} := \left(\begin{smallmatrix} u_1 & u_2 & \cdots & u_r \\ v_1 & v_2 & \cdots & v_r \end{smallmatrix}\right)$. Specifically, if w is a permutation, then $P(w) = Q(w^{-1})$ and $Q(w) = P(w^{-1})$.*

Proof. Immediate from Matrix-ball Construction, describe in Fulton's [1], Section 4.2 ■

Corollary 3.1.6. *There is a bijective correspondence between symmetric matrices with non-negative integer entries and tableaux.*

Proof. In the description of the array-matrix correspondence, swapping the rows of an array amounts to taking the transpose of its corresponding matrix. Then, by 3.1.5 (P, Q) is associated with the matrix A if and only if its transpose A^t is associated with (Q, P). Thus,

$$A = A^t \iff [(P, Q) = (Q, P)] \iff P = Q$$

and we can view the pair of identical tableau (P, P) as the tableau P itself, so the corollary follows. ■

Corollary 3.1.7. *Let n be a natural number. There is a bijective correspondence between standard tableaux P on some shape $\lambda \vdash n$ and involutions in $w \in S_n$.*

Proof. Let P be a standard tableau on some shape $\lambda \vdash n$. Then, consider the pair of tableaux (P, P). For this proof, identify (P, P) with P. By 3.1.4 there is a permutation w on n associated with P. By 3.1.6 above, there is an associated matrix $A(w)$, which is symmetric. Hence, since the permutation matrix $A(w) = A^t(w)$ is symmetric, w is an involution. ■

For the next corollary, we introduce the following notation. Say T is a tableau on some shape λ with entries in $[m]$. Then, we use the array $(1^{\alpha_1}, 2^{\alpha_2}, \cdots, m^{\alpha_m})$ to say that T has α_1 entries 1, α_2 entries 2, $\cdots$, α_m entries m.

Proposition 3.1.8. *The number of tableaux T on λ with entries $(1^{\alpha_1}, 2^{\alpha_2}, \cdots, m^{\alpha_m})$ is the same number of tableaux T' on λ with entries $(1^{\sigma(\alpha_1)}, 2^{\sigma(\alpha_2)}, \cdots, m^{\sigma(\alpha_m)})$*

Proof. Let T be a tableaux on λ with entries $(1^{\alpha_1}, 2^{\alpha_2}, \cdots, m^{\alpha_m})$. What possible matrices A could have generated T under the RSK correspondence for matrices? By the observations made in our discussion above, A has to have column sums of $\alpha_1, \alpha_2, \cdots, \alpha_m$, respectively from left-most column to the right-most column. Thus, this problem boils down to show that there are as many A's associated to T with this property that there are A''s associated to T with column sums $\sigma(\alpha_1), \sigma(\alpha_2), \cdots, \sigma(\alpha_m)$.

For simplicity, assume that σ is a transposition between the i and $i+1$. Then, write the matrix A as

$$A = \begin{bmatrix} B & C & D \end{bmatrix}$$

Here, B stands for the first $i-1$ columns, C for columns i and $i+1$, and D for the subsequent ones. It follows that the tableau P of the pair (P, Q) associated with A is given by the product $P_B P_C P_D$, where P_i stands for the insertion tableau associated with the letter i. Hence, the matrix

$$A = \begin{bmatrix} B & C & D \end{bmatrix}$$

where C' is matrix with row sums α_{i+1} and α_i. Thus, there are many matrices A associated to T with column sums $\alpha_1, \alpha_2, \cdots, \alpha_m$ to matrices A''s associated to T with column sums $\alpha_1, \alpha_2, \cdots, \alpha_{i-1}, \alpha_{i+1}, \alpha_i, \cdots, \alpha_m$.

Now, since the transpositions generate the symmetric group, this result holds for arbitrary elements $\sigma \in S_m$. ■

3.2 Permutations and other Combinatorial Identities

Since RSK behaves well under permutations, many useful and important properties of these objects can be realized through this correspondence. We list some of these in this sections. For reference, most of these results and many more can be found in Romik's [3].

We begin with some definitions.

Definition 3.2.1. *Let $w = w_1 \cdots w_n \in S_n$. We denote by $D(w)$, the descent set of w.*

$$D(w) := \{i : w_i > w_{i+1}\}$$

For instance, if $w = 769548231$, then $D(w) = \{7, 9, 5, 8, 3\}$. Also, $D(123456789) = \emptyset$, and $D(987654321) = [9]$. Analogously, we if T is a standard tableau, we define

$$D(T) := \{i : i \text{ is in some row above that of } i + 1\}$$

For example, if $T = \begin{array}{|c|c|c|c|}\hline 1&3&5&6\\\hline 2&4&7\\\cline{1-3} 8&9\\\cline{1-2}\end{array}$ as in, then $D(T) = \{1, 3, 6, 7\}$.

Theorem 3.2.2. *If $w \in S_n$ is a permutation and (P, Q) is the pair of standard tableau corresponding to it under* RSK*, then $D(w) = D(Q)$.*

Proof. Let $w = w_1 \cdots w_n$ be a permutation and let P be its associated tableau obtained by the canonical row insertion procedure. Then, if $w_i > w_{i+1}$, 2.5.3 guarantees that the box associated with the insertion of w_{i+1} will be strictly below that of w_i. Hence, this shows one containment, $D(w) \subset D(Q)$.

For the converse, if $w_i < w_{i+1}$, using 2.5.3 again, we see that that the box associated with the insertion of w_{i+1} will be weakly above that of w_i, so that $i \notin D(Q)$, which gives the reserve containment. The result follows. ∎

Proposition 3.2.3. *Let f^λ denote the number of standard tableaux on λ. Let, $d_\lambda(m)$ denote the number of tableaux with entries in $[m]$ and shape λ. Then, for $n, m \geq 1$, we have:*

$$\sum_{\lambda \vdash n} (f^\lambda)^2 = n! \quad \textit{and} \quad \sum_{\lambda \vdash n} f^\lambda d_\lambda(m) = m^n$$

Proof. The equality on the left follows immediately from 3.1.4 and the fact that there are a total of $n!$ elements $w \in S_n$. Likewise, the equality on the right follows from the fact that the number of words in the alphabet $[m]$ of length n is m^n and the *RSK* correspondence 3.1.3. ∎

Proposition 3.2.4. *If $\lfloor n/2 \rfloor$ denotes the floor function evaluated at $\frac{n}{2}$, then*

$$\sum_{\lambda \vdash n} f^\lambda = \sum_{k=0}^{\lfloor n/2 \rfloor} \frac{n!}{(n - 2k)!\, 2^k k!}$$

Proof. By 3.1.7, it suffices to show that the number of involutions in S_n equals the RHS of the above equation. Why is that the case? First, note that if w is an involution, then $A(w)$, its associated matrix, is symmetric. In particular, $w^2 = id$. Thus, the problem boils down to counting the number of permutations w such that $w^2 = id$. First, a little lemma:

Lemma 3.2.5. *For all $w \in S_n$, $w^2 = id \iff w$ is a product of disjoint 2-cycles (transpositions).*

Proof. The backwards direction is evident as the order of transpositions is always 2. For the forward direction, assume that w satisfies the assumption. Then, it is a basic fact that every permutation can be written as the product of disjoint cycles. Assume towards a contradiction that w can be written as a product of k disjoint cycles for which, at least one of them is not a 2-cycle. WLOG, let the order of such cycle be $m \geq 3$. Then, $o(w) = lcm(i_1, i_2, \cdots, i_{k-1}, m) \neq 2$, where i_j stands for the order of cycle j. In particular, $w^2 \neq id$, a contradiction. The lemma follows. ∎

Thus, to count such permutations w, we observe that, if we fix a number of 2-cycles, say k, then we need to choose $2k$ elements out of $[n]$, and, from this pool, pair the selected elements into k disjoint cycles. In particular, there are $\binom{n}{2k}$ ways to choose $2k$ elements and $\frac{2k!}{k!2^k}$ ways to pair the selected elements into 2-cycles. But note that we can only select k such that $2k \leq n$, *i.e* $k \in \{0, 1, \cdots, \lfloor n/2 \rfloor\}$. Thus, the equality follows as required. ■

Proposition 3.2.6. *Let P be a tableau on the shape λ. Let $K(P)$ be the set of words which are Knuth equivalent to $w(P)$. Then, $|K(P)| = f^\lambda$.*

Proof. Note that every w' such that $w' \equiv w(P)$ is such that $RSK(w') \leftarrow (P, Q')$, where Q is a standard tableau with entries in $[n]$. Hence, since every word w' corresponds to some (P, Q'), it in fact corresponds to Q' directly under this identification. Since fixing a P of shape λ gives f^λ choices for (P, Q'), we are done. ■

Before we present the next result, we introduce some notation.

If $w \in S_n$, we define $L(w, k)$ to be the largest number that can be achieved by summing k disjoint increasing subsequences of w. Similarly, $dc(w, k)$ to be the largest number that can be obtained by summing the length k disjoint decreasing subsequences of w. Obviously, $L(w, 1)$ is just the longest increasing subsequence of w and, similarly, $dc(w, 1)$ is just the longest decreasing subsequence of w. This definition generalizes naturally for words w instead of permutations, under the same correspondence. It is just a matter of redefining $L(w, k)$ to be the biggest number that can be obtained by summing k disjoint **weakly** increasing subsequences of w. RSK provides the following relationship between words and their respective longest increasing and decreasing subsequences:

Theorem 3.2.7 (Greene's Theorem)**.** *Let w be a word associated with the pair of tableaux (P, Q) of shape $\lambda = (\lambda_1, \cdots, \lambda_n)$ under* RSK*. Then, for all $k \in [n]$,*

$$L(w, k) = \sum_{i=1}^{k} \lambda_i \tag{1}$$

Proof. Note that a weakly increasing sequence of w is just a sequence of numbers in the tableau P arranged from left to right. But, since the fillings of a tableau are strictly increasing from top to bottom, this sequence from left to right must be taken in different columns. Conversely, every row of P can be viewed as *some* increasing subsequence of w. Hence, $L(w, 1)$ is the length of the first row, λ_1. By the same argument, the biggest number of disjoint k weakly increasing subsequences of a word w, *i.e* $L(w, k)$, can be realized by summing all of these rows, which themselves correspond to individual increasing subsequences of w. ■

An analogous results holds for the conjugate partition λ'

Theorem 3.2.8. *Let w be a word associated with the pair of tableaux (P, Q) of shape $\lambda = (\lambda_1, \cdots, \lambda_n)$ under* RSK*. Then, for all $k \in [n]$,*

$$dc(w, k) = \sum_{i=1}^{k} \lambda'_i \tag{2}$$

where λ' stands for the conjugate partition as defined in 2.1

Proof. The conjugate partition is nothing other than exchanging columns for rows. Hence, this statement is equivalent to proving that the summation of the boxes in the first k columns of λ give indeed $dc(w,k)$.

Start with a word w and the pair (P,Q) of tableaux associated with it via RSK. Note that every strict decreasing subsequence of w is just a sequence of numbers arranged from top to bottom in the tableau. Since the sequence is strictly decreasing, and the fillings of the tableau P are increasing from left to right, these numbers must be taken in different rows. Conversely, every column of P can be viewed as a decreasing subsequence of w. Hence, by now repeating the argument presented above, the result follows. ∎

Since Professor A. Golsefidy mentioned this during my presentation, here is a proof of the Erdős-Szekeres theorem using RSK

Theorem 3.2.9 (Erdős-Szekeres)**.** *Let w be a permutation of the letters in $[n^2]$. Then, it either has an increasing subsequence of length n or a decreasing subsequence of length n.*

Proof. Let (P,Q) be the pair of SYTx of shape λ associated with the permutation w. Then, $\lambda \vdash n^2$ by 3.1.3. In particular, λ must have at least n rows or n columns. If it has n columns, then $\lambda_1 = L(w,1) \geq n$ by 3.2.7. In the second case, $\lambda'_1 = dc(w,1) \geq n$, by 3.2.8. The result follows. ∎

We now introduce a few sets that are important in the study of permutations. Let $C_{n,r}$ denote the subset of permutations $w \in S_n$ such that the first $n-r$ entries form an increasing subsequence. Also, let $\Pi_{n,r}$ denote the subset of permutations $w \in S_n$ such that their first $n-r$ entries form a longest increasing subsequence of w. Formally, we have

$$C_{n,r} := \{w \in S_n : w_1 < \cdots < w_{n-r}\} \text{ and } \Pi_{n,r} := \{w \in S_n : w_1 < \cdots < w_{n-r} \text{ and } L(w,1) = n-r\}$$

Note that, $\Pi_{n,r} \subset C_{n,r}$. Also, note that, we have the following combinatorial identity:

$$|C_{n,r}| = \binom{n}{n-r} r! = \binom{n}{r} r!$$

since, choosing an element $w \in C_{n,r}$ amounts to choosing both the $n-r$ elements of $[n]$ which will form the increasing subsequence $w_1 < \cdots < w_{n-r}$ and whatever arrangement of the other r elements is left; these are $r!$ arrangements.

Denote by $C_{n,r,i}$, where $i \leq r$, the permutations $w \in C_{n,r}$ such that there are exactly i elements which are bigger than $\{w_1, \cdots, w_{n-r}\}$. These elements partition $C_{n,r}$ so that the following relationship arises:

$$\bigsqcup_{i=0}^{r} C_{n,r,i} = C_{n,r} \tag{3}$$

Denote by $RSK(C_{n,r})$ the set of pairs (P,Q) of standard young tableau associated with the permutations $w \in C_{n,r}$ via the RSK correspondence for permutations 3.1.4. Denote by $RSK(\Pi_{n,r})$, the analogous set for permutations $w \in \Pi_{n,r}$ analogously.

The elements of $RSK(C_{n,r})$ are the pairs (P,Q) on SYT on some shape $\lambda = (\lambda_1, \cdots, \lambda_k)$ such that $n \geq \lambda_1 \geq n-r$, and, moreover, the fillings of Q are such that the first $n-r$ spots in the first row are filled by $\{1, 2, \cdots, n-r\}$ from left to right (3.2.7). Similarly, the pairs of SYT (P,Q) in

$RSK(\Pi_{n,r})$ are the pairs that satisfy the latter condition **and** also satisfy the stronger condition that $\lambda_1 = n - r$, so that the condition $L(w, 1) = n - r$ is also met. Note that $RSK(C_{n,r,i})$ can be interpreted as the subset of $RSK(C_{n,r})$ such that the stronger condition $\lambda_1 = n - r + i$ is met.

We are now ready to prove the following recurrence relation for $\Pi_{n,r}$:

Proposition 3.2.10. *The following identity holds:*

$$\sum_{r=0}^{k} \binom{k}{r} |\Pi_{n,k-r}| = \binom{n}{k} k! = C_{n,k}$$

Proof. Fix k, r such that $r \leq k$. Now, consider $RSK(C_{n,k,r})$. These consist of the pairs (P, Q) of STY on the same shape such that the first row of Q consists of $n-k+r$ boxes, whose fillings of the first $n-k$ boxes are $\{1, \cdots, n-k\}$ from left to right. Consider some pair $(P, Q) \in RSK(C_{n,k,r})$. Let $\{b_1, \cdots, b_r\}$ denote the fillings of the boxes in the first row from the $(n-k+1)^{th}$ position to the $(n-k+r)^{th}$ position, in that order. Then, note that this sequence $\{b_1, \cdots, b_r\}$ completely determines how the first row of the pair $(P, Q) \in RSK(C_{n,k,r})$ is going to look like. On the other hand, we can arrive at this pair of SYT by the following procedure. Start with a $(P', Q') \in \Pi_{n,k-r}$ of same shape as that of (P, Q). This is possible since, $\lambda_1' = n - (k - r) = n - k + r$. Then, for the boxes filled with $\{n-k+1, \cdots, n-k+r\}$, identify them with the boxes labeled $\{b_1, \cdots, b_r\}$. However, for a fixed r, k, there are valid $\binom{k}{r}$ ways of labeling the boxes in $\{b_1, \cdots, b_r\}$, since you have already chosen the $n-k$ entries in the first row and there are still r to choose from the remaining $n-(n-k)$ ones. Also, for whatever picking $\{b_1, \cdots, b_r\}$, there is only one way to order them, which is increasing from left to right. Hence, we have that

$$\binom{k}{r} |\Pi_{n,k-r}| = |C_{n,k,r}|$$

Thus, using 3 and summing over all the possible values of r, we see that the desired identity holds. ■

With the developed tools, we are able to prove the following theorem about the size of $|\Pi_{n,r}|$:

Theorem 3.2.11. *Provided that $n \geq 2k$, we have*

$$|\Pi_{n,k}| = \sum_{i=0}^{k} (-1)^{k-i} \binom{k}{i} \frac{n!}{(n-i)!}$$

In order to prove the above theorem, we will introduce yet another notation involving tableaux. For $k \leq n, 0 \leq s \leq k$, let $D_{n,k,s}$ denote the set of pairs of same-shape tableaux (P, Q), where P is a SYT on $[n]$ and Q is a tableaux with entries in $[n]$ such that the first row if filled with $\{1, \cdots, n-k, a_1, \cdots, a_s, b_1, b_2, \cdots\}$ from left to right, for some $\{a_i\}_{i=1}^{s}, \{b_j\} \in ([n] \setminus [n-k])$ and $a_1 > a_2 > \cdots > a_s,\ b_1 < b_2 < ...$, and, its subsequent rows of Q follow the necessary rules of a young tableau (see 2.3 for a refresher). Note that here, Q is not necessarily a **Young Tableaux**. Certainly, if $s \geq 2$, this Q will not be a Young tableau, because of the strictly increasing condition imposed in the sequence $\{a_i\}_{i=1}^{s}$. In fact, the only thing that is stopping Q from being a young tableau is this bad sequence $\{a_i\}_{i=1}^{s}$, as, if one were to completely erase it from the first row of Q, it would turn into a young tableaux. Visually, a tableaux Q from a pair of tableaux $(P, Q) \in D_{n,k,s}$ will look like the following:

$$\begin{array}{|c|c|c|c|c|c|c|c|}\hline 1 & \cdots & n-k & a_1 & \cdots & a_s & b_1 & \cdots \\ \hline \star & \cdots & \star \\ \cline{1-3} \vdots & \cdots \\ \cline{1-2} \end{array} \tag{4}$$

The first thing to note from this definition is that $RSK(C_{n,r}) = D_{n,r,0}$. At this point, the reader might be rightfully wondering why $D_{n,k,s}$ is important at all. Other than the obvious example of $s = 0$, it might be hard to think of a way to related these **"bad"** pair of tableaux to our study of $C_{n,r}, \Pi_{n,r}$. However, $D_{n,k,s}$ aids in the understanding of the relationship $\Pi_{n,r} \subset C_{n,r}$.

Consider the set $RSK(C_{n,r}) \setminus RSK(\Pi_{n,r})$. These are precisely the pairs (P, Q) of SYTx such that the first row of Q has at least one additional entry past $(n - r)$ in the first row. That is, $[RSK(C_{n,r}) \setminus RSK(\Pi_{n,r})] \subset D_{n,r,1}$. What is left in $D_{n,r,1} \setminus [RSK(C_{n,r}) \setminus RSK(\Pi_{n,r})]$? Those pairs of tableaux (P, Q) for which Q is not a young tableaux! That is, the pairs (P, Q) where the first row of Q is filled from left to right with $1, \cdots, n - r, a_1, b_1, \cdots$, where $b_1 < a_1$. But then, viewing b_1 as another bad element in the sequence of the first row of Q, it follows that such pairs are in fact elements of $D_{n,r,2}$. Thus, we have $D_{n,r,1} \setminus [RSK(C_{n,r}) \setminus RSK(\Pi_{n,r})] \subset D_{n,r,2}$. Again, we will see that whatever is left in $D_{n,r,2}$ are those pairs (P, Q) whose $Q's$ first row is filled in such a way is filled from left to right with $1, \cdots, n - r, a_1, a_2, b_1, \cdots$, where $b_1 < a_2 < a_1$, which can be viewed as a subset of $D_{n,r,3}$. Inductively, by the same argument, for all $1 \leq s \leq r - 1$, we will have

$$D_{n,r,s} \setminus [D_{n,r,s-1} \setminus [D_{n,r,s-2} \setminus \cdots \setminus [D_{n,r,1} \setminus [RSK(C_{n,r}) \setminus RSK(\Pi_{n,r})] \cdots]] \subset D_{n,r,s+1} \tag{5}$$

Note that, if $s = r-1$ the elements of $D_{n,r,r-1} \setminus [D_{n,r,r-1} \cdots \setminus [D_{n,r,1} \setminus [RSK(C_{n,r}) \setminus RSK(\Pi_{n,r})] \cdots]]$ are those pairs (P, Q) whose $Q's$ first row is filled with $1, \cdots, n - r, a_1, \cdots, a_{r-1}, b_1$, where $b_1 < a_{r-1} < a_{r-2} < \cdots < a_1$. That is if $s = r - 1$, then 5 becomes an equality. Thus, from this discussion, we have the following relationship:

$$RSK(\Pi_{n,r}) = RSK(C_{n,r}) \setminus [D_{n,r,1} \setminus [D_{n,r,2} \setminus \cdots \setminus [D_{n,r,s-1} \setminus D_{n,r,s-1}] \cdots]] \tag{6}$$

We are now ready to prove the above theorem:

Proof of 3.2.11. It suffices to show that for $n \geq 2k$, we have:

$$|D_{n,k,s}| = \binom{k}{s} \frac{n!}{(n-k+s)!}$$

Since, using that $D_{n,k,0} = RSK(C_{n,k})$ and 6, the desired result will follow by the inclusion-exclusion principle. Thus, to prove the above equality, we will show that there is a bijective correspondence between $D_{n,k,s}$ and $RSK(C_{n,k-s}) \times A_s$, where A_s denotes the set of all possible integer sequences $\{a_j\}_{j=1}^{s} \in ([n] \setminus [n-k])$, where $a_1 > a_2 > \cdots > a_s$. This correspondence, together with 3.2.10, will suffice to prove the equality. The bijection is quite simple.

Starting from $(P, Q) \in D_{n,k,s}$, identify the relevant sequence $\{a_j\}_{j=1}^{s}$ in the first row of this Q. For this sequence, consider the unique order preserving bijection:

$$\varphi : [\{n - k + 1, \cdots, n\} \setminus \{a_j\}_{j=1}^{s}] \mapsto [n - (k - s) + 1, \cdots, n]$$

which exists since the sets have the same size. Then, define Q' to be the tableau obtained from Q by replacing every a_j in the first row with $n-k+j$, and every other element $q \in Q$ with $\varphi(q)$. Why does this produce a well defined **young tableau** Q'? It suffices to check that the fillings are increasing in every row and column of this tableau.

The first row of Q' will now have fillings from left to right $\{1, 2, \cdots, n-k+s, \varphi(b_1), \varphi(b_2), \cdots\}$. But since $b_1 > b_2 > \cdots$, φ is order preserving, and $\varphi(q) > n-k$ for all q, we have that the first row of Q' is in increasing order. Also, since all the subsequent rows of Q are row increasing and φ is order preserving, we have that the rows of Q' are increasing from left to right. Now, since $n \geq 2k \implies n-k \geq k$, we have that the second row of Q can have no more than k boxes. In particular, there are **no** elements directly below the entries in the first row Q where a_j's leave (they are single box columns). Thus, we only have to worry about the first $(n-k)^th$ columns. This is fine, since these columns in Q are already increasing in fillings so that, once we apply φ to these entries, the columns will of Q' will remain strictly increasing as required. Indeed, Q' is a young tableau and, by construction $(P, Q') \in RSK(C_{n,k-s})$.

The reserve process is identical but now we are given (P, Q') and $\{a_j\}_{j=1}^{s} \in ([n] \setminus [n-k])$. Again, we invoke the unique order-preserving bijection φ as described in the forward procedure and we now work with φ^{-1}.

Note that, φ is unique up to the given sequence $\{a_j\}_{j=1}^{s} \in ([n] \setminus [n-k])$. Changing an entry in this sequence, will change how φ acts on it. This is precisely why we need to account for all the possible pairs $(P, Q) \in D_{n,k,s}$ that will produce the same $(P, Q') \in RSK(C_{n,k-s})$, otherwise we lose the necessary information to go back to (P, Q). By encoding (P, Q') together with some $\{a_j\}_{j=1}^{s} \in ([n] \setminus [n-k])$, we know which order preserving bijection φ to choose. ■

Definition 3.2.12. *Let $w \in S_n$, we denote by $maj(w)$ the major index of the permutation w, and it is defined by*

$$maj(w) := \sum_{i \in D(w)} i$$

For example, $w = 769548231$, then $maj(w) = 7+9+5+8+3 = 32$. Also, $w(987654321) = \frac{9(9-1)}{2}$. Similarly, if T is a SYT, we may define $maj(T) = \sum_{i \in D(T)} i$. By the above theorem, we have $maj(Q(w)) = maj(w)$, where Q represents the recording tableau related to w through RSK.

Now, we will present the *q-analogue* result for 3.2.11. This result will follow from an application of the previous theorem and the following lemma:

Lemma 3.2.13. *The following identity holds:*

$$\sum_{w \in C_{n,r}} q^{maj(w^{-1})} = \prod_{j=0}^{r-1} [n-j]_q \tag{7}$$

Where $[n]_q$ is the q-anlogue number of n, namely $[n]_q = \frac{1-q^n}{1-q}$.

Proof. We will not prove this result explicitly, since it does not involve any particular use of the combinatorics exposed thus far. For the record, there are a number of ways to prove the above result. To my knowledge, the fastest and most elegant way one can go about doing so is using the theory of of P-partitions, best explained in Stanley's [2], section 4.5. ■

Theorem 3.2.14 (Garsia and Goupil [7]). *Let $w \in \Pi_{n,r}$. Provided that $n \geq 2r$, we have*

$$\sum_{w\in\Pi_{n,r}} q^{maj(w^{-1})} = \sum_{i=0}^{r}\left((-1)^{r-i}\binom{r}{i}\prod_{j=0}^{r-1}[n-j]_q\right)$$

Proof. As commented in 3.2.12, we that $maj(w) = maj(Q(w))$ whenever $Q(w)$ represents the recording tableau of w. Then, putting the latter together with 3.1.5, we have $maj(w^{-1}) = maj(Q(w^{-1})) = maj(P(w))$, where $P(w)$ is the insertion SYT obtained by the canonical procedure. Thus, we can write

$$\sum_{w\in\Pi_{n,r}} q^{maj(w^{-1})} = \sum_{(P,Q)\in RSK(\Pi_{n,r})} q^{maj(P(w))} \tag{8}$$

Again, we use inclusion-exclusion principle from 6 to get that

$$\begin{aligned}
\sum_{w\in\Pi_{n,r}} q^{maj(w^{-1})} &= \sum_{(P,Q)\in RSK(\Pi_{n,r})} q^{maj(P(w))} &(9)\\
&= \sum_{(P,Q)\in RSK(C_{n,r})} q^{maj(P(w))} - \sum_{(P,Q)\in D_{n,r,1}} q^{maj(P(w))} + \cdots + (-1)^r \sum_{(P,Q)\in D_{n,r,r}} q^{maj(P(w))} &(10)\\
&= \sum_{(P,Q)\in RSK(C_{n,r})} q^{maj(w^{-1})} - \sum_{(P,Q)\in D_{n,r,1}} q^{maj(P(w))} + \cdots + (-1)^r \sum_{(P,Q)\in D_{n,r,r}} q^{maj(P(w))} &(11)\\
&= \prod_{j=0}^{r-1}[n-j]_q - \binom{r}{1} \sum_{(P,Q')\in C_{n,r-1}} q^{maj(P(w))} + \cdots + \binom{r}{r}(-1)^r \sum_{(P,Q')\in C_{n,0}} q^{maj(P(w))} &(12)\\
&= \prod_{j=0}^{r-1}[n-j]_q - \binom{r}{1}\prod_{j=0}^{r-2}[n-j]_q + \cdots + (-1)^r\binom{r}{r}\prod_{j=0}^{-1}[n-j]_q &(13)
\end{aligned}$$

Where lines (10)-(11) follow by applying 8, (11)-(12) by applying 3.2.13 and the bijection explained in 3.2.11. By rearranging the terms in 13, we get the desired equality. ■

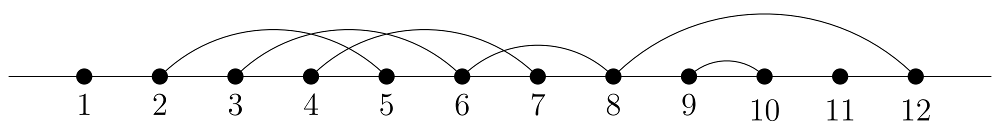
3.3 Set partitions

In this section, we will connect RSK with the theory of set partitions. The results presented in this section can be found in Chen et al [4]. Also, what follows can be primarily presented with the theory and notation we have thus far presented. However, a much more visual, and perhaps intuitive approach to the results that follow is given through the theory of **Growth Diagrams**. The latter is intrinsically related to the RSK algorithm, but, in the interest of time, we choose not to explicitly build this theory. However, the curious reader might want to refer to C. Krattenthaler's *Growth Diagrams, and Increasing and Decreasing Chains in Fillings of Ferrers Shapes* [5] for a proof of the same results using said alternative theory. We start by introducing the relevant notation for the study of set partitions.

A set partition is a way to group elements together into subsets of its own. We will be primarily interested in integer set partitions. For instance, take $S = [12] = \{1, 2, \cdots, 12\}$. Then, a set partition of S can bet taken to be $P = \{(1), (2, 5), (3, 6, 8, 12), (4, 7), (9, 10), (11)\}$, where each parenthesis is referred to as a *block* of the partition. For shorthand, we denote this partition by $25-$

368[12] − 47 − 9[10], omitting the singleton blocks whenever S is known or understood. Furthermore, another useful representation of this P, would be $[(2,5);(3,6);(6,8);(8,12);(4,7);(9,10)]$, which is called the *standard representation* of a partition, denoted by $E(P)$, and each entry $(\cdot,\cdot)$ is called an *arc* of P. For consistency, we always write the arcs of P with the left entry being the smallest of the two. It is important that the number inside the blocks of a set partition P are listed in increasing order, so that no ambiguity arises when referring to $E(P)$ or to its shorthand notation.

The standard representation of a set partition might look completely unmotivated but the parenthesis notation can be taken to visually represent the arcs of the set partition. For instance, if S, P as in our running examples, the standard representation of P can be interpreted as



We call a set partition a **complete matching on** $[2n]$, if the blocks of the set partition consists of pairs of elements. For instance, a complete matching on $S = [2(6)]$ can be taken to be $[(1,2);(3,7);(4,5);(6,8);(9,11);(10,12)]$.

We denote the space of set partitions of $[n]$ by $\mathcal{P}_n$. If $P \in \mathcal{P}_n$, we denote by $min(P), max(P)$ the set of minimal and maximal elements in each block of P respectively. Using our running example, we have $min(P) = \{1,2,3,4,9,11\}$ and $max(P) = \{1,5,12,7,10,11\}$. Note that the singleton blocks of P can be identified by the set operation $min(P) \cap max(P)$. Also, P is a complete matching on $[2n]$ if and only if $min(P) \cap max(P) = \emptyset$ **and** $min(P) \cup max(P) = [2n]$.

Let $P \in \mathcal{P}_n$. We define a $k - crossing$ of P as a k-subset $(i_1, j_1), (i_2, j_2) \cdots (i_k, j_k)$ of the arcs in the standard representation such that $i_1 < i_2 < \cdots < i_k < j_1 < j_2 < \cdots < j_k$. Hence, if P as before, then a 3-crossing of P can be taken to be the 3-subset of the arcs $(2,5),(3,6),(4,7)$. Visually, you can spot this 3-crossing by looking at how many arcs cross the arc $(2,5)$ and where they land. Similarly, we define a $k - nesting$ to be k-subset $(i_1, j_1), (i_2, j_2) \cdots (i_k, j_k)$ of the arcs in the standard representation such that $i_1 < i_2 < \cdots < i_k < j_k < j_{k-1} < \cdots < j_1$. In our running example, a 2-nesting of P can be taken to be the arcs $(8,12),(9,10)$. Visually, you look for a long arc such that other arcs are nested inside of it. We denote by $cr(P)$ (resp., $ne(P)$) the maximal number k such that P has a k-crossing (resp, k-nesting).

Now, we will connect the study of these statistics of set partitions to the study of tableaux. First, a definition:

Definition 3.3.1. *We call a* ***vacillating tableaux*** *of shape λ and length $2n$, denoted by V_λ^{2n}, a walk in the Young's Lattice (2.2), starting from the empty shape $\emptyset$ and arriving at λ at the $2n^{th}$ step, wherein each step follow these specific rules:*

- *Do nothing twice (i); OR*
- *Do nothing and then add a square (ii); OR*
- *Remove a square and then do nothing (iii); OR*
- *Remove a square and then add a square (iv)*

In other words, you can view this walk as being determined by whatever strategy you choose at each odd step. However, it is important to note that there are strategies which share the same first step, so a strategy can be completely observed only after every 2 steps. Another way we can represent the vacillating tableau V_λ^{2n} is by the sequence of shapes λ^i that are visited in the corresponding walk through the Young's Lattice.

We choose to represent a $2n$ walk in the Young's lattice by the array $(\lambda^0, \lambda^1, \cdots, \lambda^{2n})$, and we can view $(\emptyset = \lambda^0, \lambda^1, \cdots, \lambda^{2n} = \lambda)$ as an element V_λ^{2n}, provided that the walk follows the rules above. Here, it is important to note that, in order to avoid confusion, we use the superscript notation to denote the i^{th} shape of visited by the walk, which is different from the i^{th} entry of the partition λ, thus far denoted by λ_i.

We now give four examples of a vacillating tableaux of the form V_2^6.

$$\begin{array}{ccccccc} \lambda^0 & \lambda^1 & \lambda^2 & \lambda^3 & \lambda^4 & \lambda^5 & \lambda^6 \\ \emptyset & \emptyset & 1 & 1 & 2 & 2 & 2 \\ \emptyset & \emptyset & \emptyset & \emptyset & 1 & 1 & 2 \\ \emptyset & \emptyset & 1 & \emptyset & 1 & 1 & 2 \\ \emptyset & \emptyset & 1 & 11 & 1 & 2 & 2 \end{array} \tag{14}$$

A couple of observations are due. Firstly, note that we can only have $\lambda^{i-1} \subset \lambda^i$, whenever $i = 2k$ for some $k \in [n]$. This is because the only strategies that allow us to add a square are (ii) and (iv); both of which tell us to add a square at the even step. Similarly, looking at (iii) and (iv), we can conclude that, if $\lambda^{i-1} \supset \lambda^i$, then $i = 2k - 1$ for some $i \in [n]$. Nothing specific can be said for the case that $\lambda^{i-1} = \lambda^i$, since you can remain in place at any point during the walk, with any given strategy.

Using the RSK algorithm, we will now construct a bijection between vacillating tableaux on the empty shape and of size $2n$, $i.e$ $V_\emptyset^{2n}$, and $P \in \mathcal{P}_n$.

Let $P \in \mathcal{P}_n$ and $E(P)$ be its standard representation. We will construct a sequence $\{T_i\}_{i=0}^{2n}$ of SYTx inductively, such that, the shape λ^i of T_i will be determined uniquely by the arcs of the standard representation of P. Let $T_{2n} = \emptyset$, then for each $j \in [n]$ one by one from n to 1, T_{2j-1} and T_{2j-2} are determined as follows:

- If j is an singleton block, then we set $T_{2j-1} = T_{2j-2} = T_{2j}$
- If j is a right end-point of an arc (i, j) but **not** a left end-point of an arc (j, k), then we set $T_{2j-1} = T_{2j}$ and $T_{2j-2} = T_{2j} \leftarrow i$.
- If j is a right end-point of an arc (i, j) **and** a left end-point of an arc (j, k), then T_{2j-1} is obtained from T_{2j} by deleting j and we set $T_{2j-2} = T_{2j-1} \leftarrow i$.
- If j is a left end-point of an arc (j, k) but **not** a right end-point of any arc (i, j), then T_{2j-1} is obtained from T_j by deleting j and we set $T_{2j-2} = T_{2j-1}$.

Note that, since $T_{2n} = \emptyset$, we can only remove entries that were once inserted. Also, at each $j \in [n]$, we can only add entries to T_{2j-1} which are strictly smaller than j. Thus, at each $j \in [n]$, T_{2j} can only have entries which are as big as k. Indeed, then, deleting the entry j from T_{2j} would only amount to deleting an outside corner. Hence, this process produces a well defined sequence of standard tableaux $\{T_i\}_{i=0}^{2n}$. Also, note that for each i we insert (associated with the arc (i,j)), there exists $j' \in [n]$, where $j' < j$ such that i is an entry of $T_{2j'-1}$. Hence, all entries are eventually removed and $T_0 = \emptyset$. Indeed, then, the sequence $\{\lambda^i\}_{i=0}^{2n}$ of shapes associated with $\{T_i\}_{i=0}^{2n}$ is as a vacillating tableaux.

But why is $\{\lambda^i\}_{i=0}^{2n}$ **uniquely** determined by $\{T_i\}_{i=0}^{2n}$? Because the representation $E(P)$ is unique! To elaborate on this, note that if we could get the same $\{\lambda^i\}_{i=0}^{2n}$ for two different sequences $\{T_i'\}_{i=0}^{2n}$ and $\{T_i\}_{i=0}^{2n}$ they would have to agree at each deletion step of the walk. However, note that the deletion step is completely determined by the entries i inserted, since we only delete those entries which we have inserted and, eventually, we delete them all. In turn, these inserted $i's$ are unambiguously determined by the arcs (i,j) in the standard representation $E(P)$! Thus, in fact, $\{T_i'\}_{i=0}^{2n}$ and $\{T_i\}_{i=0}^{2n}$ would have to be produced from the same set partition P. We denote the map $P \to \{\lambda^i\}_{i=0}^{2n}$ by ψ. Next, to show that this is a bijection, we produce the inverse process.

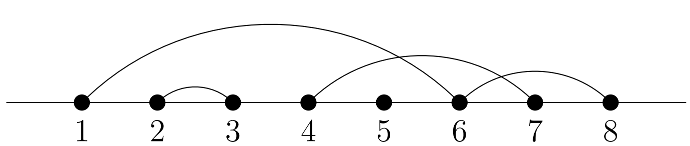
Let $(\emptyset = \lambda^0, \lambda^1, \cdots, \lambda^{2n} = \emptyset) \in V_\emptyset^{2n}$. Inductively, we produce two sequences $\{P^i\}_{i=0}^{2n}$ and $\{T_i\}_{i=0}^{2n}$ of set partitions and of SYTx respectively. Let P^0 be the empty set and T_0 be the tableaux on the empty shape. Then, for each $i \in [2n]$, we define

- If $\lambda^{i-1} = \lambda^i$, then $P^i = P^{i-1}$ and $T_i = T_{i-1}$
- If $\lambda^{i-1} \subset \lambda^i$, then, as remarked, there exists $k \in [n]$ such that $i = 2k$. We set $P^i = P^{i-1}$ as before, but we let T_i be the unique tableaux on the shape λ^i/λ^{i-1} obtained by filling the new box with the entry k.
- If $\lambda^{i-1} \supset \lambda^i$, then i, there exists $k \in [n]$ such that $i = 2k - 1$. Using the fact that RSK is completely reversible process, let T_i be the unique tableau of shape λ^i such that $T_i \leftarrow j = T_{i-1}$ for some j. At this point, note that $j < k$ since you have not yet inserted anything bigger than $k - 1$, provided that $i = 2k - 1$. In this case, P_i is obtained from P_{i-1} by adding the arc (j,k) to its standard representation.

We set $P := P^{2n}$ under this procedure and we denote the map $(\emptyset = \lambda^0, \lambda^1, \cdots, \lambda^{2n} = \emptyset) \to P$ by ϕ, or $\phi(\{\lambda^i\}_{i=0}^{2n}) = P$. Using much of the same argument as for the last algorithm, one can show that $\{T_i\}_{i=0}^{2n}$ is indeed a well defined sequence of STYx such that $T_0 = \emptyset = T_{2n}$ uniquely determined by the vacillating tableaux $\{\lambda^i\}_{i=0}^{2n}$. In fact, $\{T_i\}_{i=0}^{2n}$ is of little important to this map; we only care about it insofar it helps us prove that ϕ and ψ are inverses.

Every element $(\emptyset = \lambda^0, \lambda^1, \cdots, \lambda^{2n} = \emptyset) \in V_\emptyset^{2n}$ produces a unique sequence of SYTx $\{T_i\}_{i=0}^{2n}$ and some $P \in \mathcal{P}_n$ under ψ. However, we constructed this algorithm in such a way that whenever we run P through by ϕ, we will get $\{T_i\}_{i=0}^{2n}$. But, as argued, under ψ, $\{T_i\}_{i=0}^{2n}$ completely determines the vacillating tableau $\{\lambda^i\}_{i=0}^{2n}$. Thus, we have $\psi \circ \phi(P) = P$ and $\phi \circ \psi(\{\lambda^i\}_{i=0}^{2n}) = \{\lambda^i\}_{i=0}^{2n}$, showing that ϕ and ψ are indeed inverses of each other.

Example 3.3.2. *Let $P = 168 - 23 - 47$ a set partition of $[8]$. Then it's standard representation is:*



We first run the backwards algorithm ψ *to get a vacillating tableau* $\{T_i\}_{i=0}^{16}$. *Starting from* $\emptyset$ *on the right, we go from 8 to 1 in 8 steps. (1) Do nothing, then using RSK insert 6, (2) Do nothing, then using RSK insert 4, (3) Delete 6, then insert 1, (4) do nothing twice, (5) delete 4, then do nothing, (6) do nothing, then insert 2, (7) delete 2, then do nothing, (8) delete 1, then do nothing. The corresponding* $\{T_i\}_{i=0}^{16}$ *will be:*

$$\emptyset, \emptyset, \boxed{1}, \boxed{1}, \begin{array}{|c|c|}\hline 1 & 2 \\ \hline\end{array}, \boxed{1}, \boxed{1}, \boxed{1}, \begin{array}{|c|}\hline 1 \\ \hline 4 \\ \hline\end{array}, \begin{array}{|c|}\hline 1 \\ \hline 4 \\ \hline\end{array}, \begin{array}{|c|}\hline 1 \\ \hline 4 \\ \hline\end{array}, \boxed{4}, \begin{array}{|c|}\hline 4 \\ \hline 6 \\ \hline\end{array}, \boxed{6}, \boxed{6}, \emptyset, \emptyset$$

Which, as argued, uniquely corresponds to the vacillating tableau $\{\lambda^i\}_{i=0}^{16}$

$$\emptyset, \emptyset, 1, 1, 2, 1, 1, 1, 11, 11, 11, 1, 11, 1, 1, \emptyset, \emptyset$$

As a consistency check, we will now perform ϕ *on this resulting vacillating tableaux. Starting from* $\emptyset = T_0$, *we will construct* $\{T_i\}_{i=0}^{16}$ *in 16 steps. (1) Do nothing. (2) Since* $\lambda^1 \subset \lambda^2$, *fill the new box with entry* $\frac{2}{2} = 1$. *(3) Do nothing. (4) Since* $\lambda^3 \subset \lambda^4$, *fill the new box with entry* $\frac{4}{2} = 2$. *(5)* $\lambda^4 \supset \lambda^5$, *then remove* 2 *from your tableau. Why? We currently have* $\begin{array}{|c|c|}\hline 1 & 2 \\ \hline\end{array}$*; Thus,* 2 *is the unique entry of the current running tableau such that removing it and inserting in the resulting tableau will actually produce our current tableau. (6)-(7) Do nothing. (8) Since* $\lambda^7 \subset \lambda^8$, *fill the new box with entry 4. (9)-(10) Do nothing. (11) Using the same reasoning as in step 5, remove* 1. *Since* $\lambda^{11} \subset \lambda^{12}$, *fill the new box with entry 6. (13) Using the same reasoning as in step 5, remove* 4. *(14) Do nothing. (15) Using the same reasoning as in step 5, remove* 6. *(16) Do nothing.*

Throughout this process, we removed the entries $2, 1, 4, 6$ *at the* $5^{th}, 11^{th}, 13^{th}, 15^{th}$ *steps respectively. Noting that* $5 = 2(3) - 1$, $11 = 2(6) - 1$, $13 = 2(7) - 1$, $16 = 2(8) - 1$, ϕ *tells us that* $P = \{(2, 3), (1, 6), (4, 7), (6, 8)\}$ *as required.*

From doing these examples, one can get a feel as to how this algorithm relates to the study of crossings and nestings. Notice the pattern through which entries make their way into and leave the sequence tableaux $\{T_i\}_{i=0}^{16}$ above. In this example, 1 comes in at T_2 and stays in the sequence of tableaux up until T_{10}, 2 comes in at T_4 and leaves at the same step, 4 comes in at T_8 and leaves at T_{12}, 6 comes in at T_{12} and leaves at T_{14}. The pattern here is that entries i are entering at the tableaux $T_2 i$ and leaving at the tableau T_{2j-2}, where (i, j). This is not a coincidence. We built the algorithm to add left hand points and remove right end points deterministically. In fact, if $\{T_i\}_{i=0}^{2n}$ is a sequence of tableaux associated with the vacillating tableaux $\{\lambda^i\}_{i=0}^{2n}$ via $\psi(P)$, then (i, j) is an arc of P if and only if i is an entry on of the subsequence $\{T_k\}_{k=2i}^{2j-2}$. If this is the case, we say that the integer i is added to the sequence $\{T_i\}_{i=0}^{2n}$ at step i and leaves at step j.

The latter fact motivates the following proposition.

Proposition 3.3.3 ([4]). *Let* $P \in \mathcal{P}_n$ *and* $\{T_i\}_{i=0}^{2n}$ *denote the associated sequence of tableau obtained by* $\psi(P)$. *Then, the arcs* $(i_1, j_1), \cdots, (i_k, j_k)$ *form a k-nesting of* P *if and only if there exists a tableau* $T_l \in \psi(P)$ *such that* $i_1, \cdots, i_k \in content(T_l)$ *and they leave* $\{T_i\}_{i=0}^{2n}$ *in decreasing order. Similarly, the arcs* $(i_1, j_1), \cdots, (i_k, j_k)$ *form a k-crossing of* P *if and only if there exists a tableau* $T_l \in \psi(P)$ *such that* $i_1, \cdots, i_k \in content(T_l)$ *and they leave* $\{T_i\}_{i=0}^{2n}$ *in increasing order.*

Proof. Since the authors proved this for crossings only, we offer a proof of this fact for nestings and refer the interested reader to the aforementioned paper. However, we note that the exact same idea works for crossings.

Assume that the arcs in questions form a $k - nesting$, such that $i_r < j_r$, for all $1 \leq r \leq k$ and $i_1 < \cdots < i_k < j_k < \cdots < j_1$. Then, as remarked above, $i_1, \cdots, i_k$ enter at steps $i_1, \cdots, i_k$ and leave at steps $j_1, \cdots, j_k$ respectively. But, by assumption, $j_k < \cdots < j_1$, so that i_k is the first to leave, i_{k-1} the second, and, inductively, i_1 is the last to leave. Indeed, $T_l = T_{2(i_k)}$ and they leave in decreasing order as required.

For the converse, if there are k integers $i_1 < \cdots < i_k$ leaving in decreasing order from $\psi(P)$ at steps $j_k < \cdots < j_1$ respectively, then P has arcs $(i_1, j_1), \cdots, (i_k, j_k)$, forming a k-nesting subset of P as required. ■

Theorem 3.3.4 (Thm 3.2 [4]). *Let $P \in P_n$ and $\psi(P) = (\emptyset = \lambda^0, \lambda^1, \cdots, \lambda^{2n} = \emptyset)$. Then, $cr(P)$ is the most number of rows in any λ^i, and, similarly, $ne(P)$ is the most number of columns in any λ^i.*

In order to prove this theorem, one needs to understand how the characterization of k-crossings and k-nestings presented in 3.3.3 affects the shape of the sequence $\psi(P)$. To that end, we construct a sequence of permutations $\{\sigma_i\}_{i=0}^{2n}$ acting on the *content*(T_i) backwards inductively as follows. Let $\sigma_{2n} = \emptyset$. As usual, denote by λ^i the associated shape of T_i. For each T_{i-1} in the associated sequence of $\psi(P)$

- If $T_i = T_{i-1}$, define $\sigma_{i-1} := \sigma_i$.
- If $\lambda^i \subset \lambda^{i-1}$, then as argued before, observing T_{i-1} and T_i, we can determine the unique j such that $T_i \leftarrow j = T_{i-1}$. In this case, append j at the end of σ_i, *i.e* $\sigma_{i-1} = \sigma_i j$.
- If $\lambda^i \supset \lambda^{i-1}$, then $i = 2k$ for some $k \in [n]$, and T_i/T_{i-1} is a skew tableaux with a single box filled with the entry k. In that case, not only k must be a letter in σ_i, but also must be the biggest one. Let σ_{i-1} be the permutation obtained from σ_i by simply deleting the entry k.

Note that at any point $i \in [2n]$, σ_i is able to record both the content of T_i and at which order these entries will exit T_i. That is, if $\sigma_i = r_1 r_2 \cdots r_k$, then since r_k was the last appended to σ_i backwards inductively, it is the first one to leave, and similarly, r_{k-1} is the second one to leave, and so on, all the way up to r_1, which leaves last.

The following claim is the remaining piece of the puzzle so that we can finally prove 3.3.4.

Claim 3.3.5. *Let Let $P \in P_n$ and $\{T_i\}_{i=0}^{2n}$ be obtained via $\psi(P)$. Let $\{\sigma_i\}_{i=0}^{2n}$ be obtained by the above procedure. Then, let $(A(\sigma_i), B(\sigma_i))$ be the associated pair of SYT to σ_i via the RSK algorithm. Then, the recording tableau $A(\sigma_i)$ agrees with T_i,* i.e *$A(\sigma_i) = T_i$.*

Proof. This is proven by backwards induction on the sequence $\{\sigma_i\}_{i=0}^{2n}$. Clearly, since $T_{2n} = \emptyset = \sigma_{2n}$, the base case holds. Now, let $i \in [2n]$ and assume that the result is true for all $i \geq j \in [2n]$. If $T_{i-1} = T_i$, then $\sigma_i = \sigma_{i-1}$ and the result holds by the induction hypothesis. Now, If $\lambda^i \subset \lambda^{i-1}$, then we can determine the unique j such that $T_i \leftarrow j = T_{i-1}$. In this case, $\sigma_{i-1} = \sigma_i j$. But since, $T_i = A(\sigma_i)$, then,

$$A(\sigma_{i-1}) = A(\sigma_i j) = A(\sigma_i) \leftarrow j = T_i \leftarrow j = T_{i-1}$$

If $\lambda^i \supset \lambda^{i-1}$, then $i = 2k$ for some $k \in [n]$, and T_i/T_{i-1} and σ_{i-1} is the permutation obtained from σ_i by deleting the entry k, which is the largest at that point. Note that since k is the largest entry of σ_i, it is an outside corner of T_i. If k is in the first row, it is the right-most box of T_i, so that the result holds trivially. We omit the proof for the other cases but much of the same idea works, one just have to tediously make an induction argument for every row above of the one considered. ■

Proof of Theorem 3.3.4. To prove this, we will put together 3.3.3 and the above claim. Let $\{T_i\}_{i=0}^{2n}$ be obtained via $\psi(P)$. Firstly, recall that 3.3.3 states that P has a k-nesting if and only if there exists a tableau $T_l \in \psi(P)$ such that $i_1, \cdots, i_k \in content(T_l)$ and they leave the sequence of tableaux in decreasing order. Thus, because of the way we constructed $\{\sigma_i\}_{i=0}^{2n}$, P has a k-nesting (resp. k-crossing) if and only if there is an associated σ_l with an increasing subsequence of length k (resp. decreasing subsequence of length k).

Also, by the claim above, $A(\sigma_l) = T_l$. But, Greene's Theorem 3.2.7 tells us that the number of boxes in the first row (resp, first column) of the associated partition of $A(\sigma_l)$, which is λ^l, reveals the length of σ_l's longest increasing subsequence (resp. longest decreasing subsequence). Hence, P has a k-nesting (respect. k-crossing) if and only if there exists a $T_l \in \{T_i\}_{i=0}^{2n}$, with k boxes in the first row (resp, k boxes in the first column). Hence, result follows as required. ■

Proposition 3.3.6 (Prop 3.4 [4]). *Let $P \in \mathcal{P}_n$ and consider its standard representation*

$$E(P) = [(i_1, j_1), (i_2, j_2), \cdots, (i_k, j_k)]$$

arranged such that $j_1 < j_2 < \cdots < j_k$. Define $\alpha(P) = i_1 \cdots i_k$. Then, the set of nestings of P is in bijection with the set of decreasing subsequences of $\alpha(P)$.

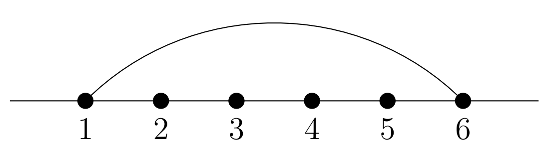
Proof. Let $(\tilde{i}_1, \tilde{j}_1), \cdots, (\tilde{i}_r, \tilde{j}_r)$ be a r-nesting of P where $\tilde{j}_1 < \cdots < \tilde{j}_r$. Then, since this is a r-nesting, we must have $\tilde{i}_r < \cdots < \tilde{i}_1$, so that $\tilde{i}_1, \cdots, \tilde{i}_r$ forms a decreasing subsequence of $\alpha(P)$.

Now, let $\tilde{i}_1, \cdots, \tilde{i}_r$ be a r-decreasing subsequence of $\alpha(P)$. Consider $(\tilde{i}_1, \tilde{j}_1), \cdots, (\tilde{i}_r, \tilde{j}_r)$. Since $\tilde{j}_1 < \cdots < \tilde{j}_r$ and, if $\{\sigma_i\}_{i=0}^{2n}$ is the associated sequence via $\psi(P)$, By 3.3.3, there exists an associated σ_l where the sequence $\tilde{i}_r \tilde{i}_{r-1} \tilde{i}_{r-2} \cdots \tilde{i}_1$ is part of it. Thus, these entries leave $\{T_i\}_{i=0}^{2n}$ in reverse order. Indeed, they represent a r-nesting of P as required. ■

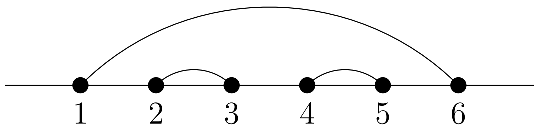
Unlike many (if not all) of the results presented in this section, this last proposition does not have an analogue for crossings of P. Both the authors [4] and [5] report that they have not been able to arrive at a r-crossing analogue for 3.3.6. I have spent a great deal of this past year thinking about how to extend this proposition to crossings, but, at this point, I am not sure if I have made any progress. However, I do believe that I understand this problem greatly. We will now discuss about the shortcomings of this generalization. Hence, we dedicate the next subsection of this report to a brief discussion of this problem

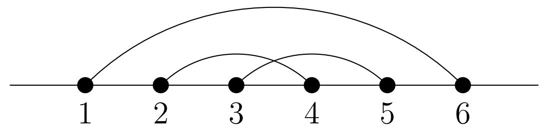
3.4 Thoughts on the analogue for r-crossings

Understanding this problem boils down to understanding how crossings and nestings are inherently different in their structure. I like to say that nestings are **robust** structures, whereas crossings are **fragile**. Why? Say we start with a simple partition P of 6 where every number is inside their own block, but 1 and 6, which share a block. It standard representation is:

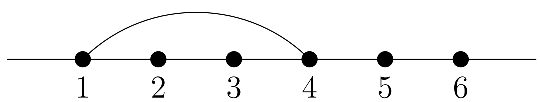


Note that we have 4 available spots to put arcs on. One can check that no matter how we decide to put arcs on this standard representation (provided that they follow the rules of doing so) I am guaranteed to have at least a 2-nesting. Here are a couple of options:

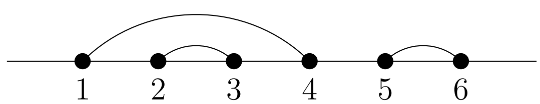




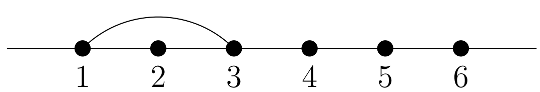
In both of these options, I have #2 2-nestings of P. But, now let's look at a similar situation. Here we start with a different partition P' of 6. Its standard representation is:



Now, I have the same number of available spots to put arcs around, but there are fillings that do not increase the number of crossings. For instance,

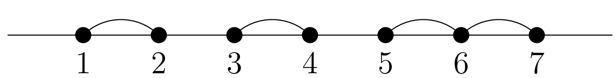


One might think that this was just an occasional example, but in fact, even upon choosing the unfriendliest partition to add a nesting, say:



one could still just leave 2 alone and add other arcs which will not create crossings. So, in a sense, nestings are easier to create. That is, you can always guarantee them under certain conditions. This is partly the reason why 3.3.6 is hard to extend.

Certainly, any r-crossing of P corresponds to an r-increasing subsequence of $\alpha(P)$. However, we are not guaranteed that every r-increasing subsequence of $\alpha(P)$ corresponds to a $r - crossing$. To see this, think about the following partition of 7: $E(P) = [(1,2),(3,4),(5,6),(6,7)]$. Then $\alpha(P) = 1356$, which is itself a strictly increasing word, yet, P has no crossings.



This anomaly can be countered by imposing the condition that the increasing subsequence $\tilde{i}_1, \cdots, \tilde{i}_r$ is such that $\tilde{i}_r < \tilde{j}_1$, so that all arcs are guaranteed to intersect the arc $(\tilde{i}_1, \tilde{j}_1)$. This is a bit unfortunate because then we have a less interesting statement. We would like a statement that is independent of the endpoint of arcs.

One way we to word this lesser interesting statement is by defining $\alpha(P)$ in terms of endpoints:

Proposition 3.4.1. *Let $P \in \mathcal{P}_n$ and consider its standard representation*

$$E(P) = [(i_1, j_1), (i_2, j_2), \cdots, (i_k, j_k)]$$

arranged such that $i_1 < i_2 < \cdots < i_k$. Define $\alpha(P) = j_1 \cdots j_k$. Then, if there exists $n < k$ such that $i_k < j_n$, then, any r-increasing subsequence of $\alpha(P)$ which has starting point weakly after j_n corresponds to a r-crossing of P.

Proof. Immediate from the previous discussion. ■

Although 3.4.1 is still not the result one would want as far as a r-crossing analogue it does give us some insight on our original discussion about how to draw arcs such that we arrive at a crossing. Notice that, the counterfeit solution proposed in 3.4.1 consists in finding some $n < k$ such that $i_k < j_n$ and then considering increasing subsequences which start weakly after j_n. A couple of observations then follow.

Notice that, if n is too big (close to k), then we have fewer possibilities to find r-long increasing subsequences of $\alpha(P) = j_1 \cdots j_n \cdots j_k$. In fact, the best we could do in this case would be a $k - n$ long crossing. On the other hand, if we couldn't observe the choices for $i_1, \cdots, i_k$, but could still choose our r-increasing sequence $j_{k_1} < \cdots < j_{k_r}$, we would not want to choose j_{k_1} so small that we we would fail to achieve arcs that intersect, just as in the deranged example given above. Thus, if we look at this as a game, where our goal is to create crossings without complete information, it is quite unclear which strategies to adopt. Visually, this is because we want arcs big enough so that we can still draw other arcs whose left end-points are inside of it, but we don't want it so big that we cannot connect the right end-points of these arcs outside of this graph.

Now, for nestings, there is no such ambiguity: We are always better off having big right endpoints and small left hand points. Visually, we want to envelope as many points under an arc so that we still allow for the possibility of nestings inside such. In fact, this is precisely the way to maximize nestings; that is, choosing arcs such that $i_1 < i_2 \cdots < i_k < j_k < j_{k-1} < \cdots < j_k$, we have a k crossing.

The dicothemy between these two justifies our labeling of crossings and nestings as being fragile and robust respectively. Even though this line of argument would support the idea of those who think that there is no such analogue for r-crossings, there is still a sense in which this would be deeply unsatisfying. Why? Because, from the results already known, these objects are highly symmetric. For instance, the following is one of the main theorems of [4]:

Theorem 3.4.2 (Theorem 1.1 [4])**.** *Let $S, T \subset [n]$. Denote by $\mathcal{P}_n(S, T)$ the subset of $\mathcal{P}_n$ where the partitions P are such that $min(P) = S$ and $max(P) = T$. Then,*

$$\sum_{P \in \mathcal{P}_n(S,T)} x^{cr(P)} y^{ne(P)} = \sum_{P \in \mathcal{P}_n(S,T)} x^{ne(P)} y^{cr(P)} \tag{15}$$

We are not going to prove this theorem, but the proof basically revolves around applying the results found in 3.3.4. What is interesting here is that the equality is essentially saying that the arithmetic statistics $cr(P), ne(P)$ have symmetric joint distributions over $\mathcal{P}_n(S, T)$. So, in fact, when we are talking about maximality, nestings and crossings are essentially just as frequent.

Another example which portray these objects' co-symmetry would be the number of non-crossing and non-nesting partitions. For instance, one can easily show that non-crossing set partitions of $[n]$ are Catalan objects.

Proposition 3.4.3. *Let $NC(n)$ determine the number of non-crossing partitions of $[n]$, where we define $NC(0) = 1$ (the empty partition is always non-crossing). Then, $NC(n)$ satisfy the following recurrence relation for $n \geq 0$*

$$NC(n+1) = \sum_{k=0}^{n} NC(k)NC(n-k) \tag{16}$$

Proof. Consider the set $[n+1]$ and some set partition of it, say P. Then, $[n+1]$ must be in some block of this partition. Let k be the minimum number inside this block. Then, first note that we cannot have any arc of the form (i, j) where $i \leq k-1$ and $j \geq k+1$, otherwise we would already have a crossing (since k and $n+1$ are connected. Thus, if we would like to count the number of non-crossings, we have to rule out this possibility. We do that by requiring that the number $\{1, \cdots, k-1\}$ are inside form their own non-crossing partition. How many ways can we do that? Precisely $N(k-1)$.

Now, the same logic applies to the numbers bigger than k, since we cannot connect any of them to the ones smaller than k (by the argument presented above). Thus, in fact, we may then look at the numbers $\{k, \cdots, n+1\}$ as a non-crossing set partition of $n-k+1$ numbers. There are precisely $NC(n-k+1)$ of these. In particular, we have

$$NC(n+1) = \sum_{k=1}^{n} NC(k-1)NC(n-k+1) = \sum_{k=0}^{n} NC(k)NC(n-k) \tag{17}$$

Hence, $NC(n)$ satisfies the Catalan recursion and the proposition follows. ■

Much of the same argument can be used to prove that the number of non-nesting partitions of $[n]$ is also Catalan.

This recurrence relation for non-crossing and non-nesting is not new at all, but for a long time, mathematicians couldn't come up with a bijection between them. In 2007, D. Panyushev discovered a explicit bijection between these two sets. Unfortunately, this bijection is not simple and it goes beyond the theme of this report, so we will not present it. Its complexity also does not help in understanding the discussed problem of r-crossings (at least, not that I can tell). For the interest reader, I recommend the paper [6], where the authors present Panyushev's explicit bijection and construct yet another one. Thus, 3.4.3 above shows that, when it comes down to completely eradicating the occurrences of crossings or nestings, we do achieve symmetry, just as in 3.4.2.

We have now seen two examples on the symmetric properties of crossings and nestings. Do these contradict the intuition developed at the beginning of our discussion? Not at all. Why? Because 3.4.2 is about maximality, 3.4.3 is about minimality, and what we are trying to investigate are occurrences! That is, we are trying to find a way that $\alpha(P)$ records every r-crossing, and not just the biggest or the smallest. Now, there is also a sense in which maximality and minimality should play an important role in occurrences, after all, they are the maximum and minimum number of occurrences respectively. But again, if this is the case, then it is not outwardly trivial.

I have some more thoughts and useful insights about this problem using the theory of Growth Diagrams, shown in [5]. Since we have not explicitly built this theory here, I will say no more about extending this proposition. If the reader is interested in this problem, don't hesitate to contact me.

4 Applications to the Representation Theory of Symmetric Polynomials and Functions

Representation Theory is a field of math generally concerned with abstract algebraic structures. As one would expect, the process of grasping these complex structures is hard on its own. Representation theorists try to reduce this problem by studying modules over these structures, representing their elements as linear transformations of vector spaces.

In this report, we will be mainly concerned with the theory of symmetric functions and, particularly, how the combinatorics of Young Tableaux may help us understand and solve problems related to such objects.

We will start with a brief introduction to the classical theory symmetric polynomials and introduce our main object of study in this section, the Schur polynomials. Then, we will formally construct the ring of symmetric functions and investigate how this theory agrees with the theory generalizes the theory of symmetric polynomials. Our goal in this section is to show how the combinatorics of Young Tableaux relates to this theory and how it is useful in proving different results regarding Schur Polynomials, most notably the Littlewood-Richardson coefficients, which provides a combinatorial formula to compute product of these specific family of symmetric polynomials.

It is important to keep in mind that there are many different ways that one can develop the theory of symmetric functions, *viz.*, combinatorially using power series, or by purely algebraic means. Since this report in intended for a wider audience than the veteran mathematician, we choose to present this topic using knowledge of power series because this approach is most closely related to combinatorics. However, of course, all roads lead to Rome and any approach produces the same theory. For reference, we will primarily adopt the presentation of this topic exposed in Stanley's [2], *Chapter 7*. However, less experienced readers may find the presentation of this topic in Sagan's **??** easier to chew on.

4.1 Symmetric Polynomials

As the name suggests, symmetric polynomials are nothing other than polynomials that are invariant under a permutation of its variables. To make this precise, first consider the ring of polynomials in n variables with coefficients in a commutative ring R. That is, consider $R[x_1, \cdots, x_n]$. Then, we define an action of S_n on R via:

$$\sigma \cdot f(x_1, \cdots, x_n) = f(x_{\sigma_1}, \cdots, x_{\sigma_n})$$

where $\sigma \in S_n$ and $f \in R[x_1, \cdots, x_n]$. With this in mind, we call f a **symmetric polynomial** over R whenever $\sigma \cdot f = f$. When R is understood we may refer to f as a symmetric polynomial only. In this report, we will mainly be concerned with polynomial rings over $\mathbb{Z}$ or $\mathbb{Q}$. We denote the space of symmetric polynomials (or functions) in n variables over $\mathbb{Z}$ by Λ_n. Note that $(\Lambda_n, +, \cdot)$ is a subring of $\mathbb{Z}[x_1, \cdots, x_n]$ with the usual polynomial addition and multiplication.

Inside this ring, there are a couple of special symmetric polynomials that earn distinguishable names. Let $\lambda = (\lambda_1, \cdots, \lambda_n)$, then, we define the *monomial symmetric polynomial* associated with λ to be the polynomial

$$m_\lambda(x_1, \cdots, x_n) := \sum_{\sigma \in S_n} x_{\sigma(1)}^{\lambda_1} \cdots x_{\sigma(n)}^{\lambda_n}$$

Alternatively, this is shorthanded by the following notation. If λ as above, then we denote $x^\lambda := \prod_{i=1}^{l(\lambda)} x_i^{\lambda_i}$, and if with this new notation, we have

$$m_\lambda(x_1, \cdots, x_n) = \sum_{\sigma \in S_n} x^{\sigma(\lambda)}$$

where σ acts on λ by permuting the indexes of it. By definition, we see that m_λ is a symmetric polynomial.

Similarly, for $i \in [n]$, we denote by $e_j(x_1, \cdots, x_n)$ the j^{th} *elementary symmetric polynomial* on n variables. It is defined in terms of the monomial symmetric polynomial associated with the associated partition $1^j = (1, \cdots, 1)$ representing the tableau with a single column of j boxes:

$$e_j(x_1, \cdots, x_n) := m_{1^j}(x_1, \cdots, x_n) = \sum_{1 \le i_1 < i_2 < \cdots < i_j \le n} x_{i_1} \cdots x_{i_j} \tag{18}$$

We may abbreviate $e_j(x_1, \cdots, x_n) =: e_j^{(n)}$. For all $1 \le j \le n$, the following recursive identity for elementary symmetric polynomials hold:

$$e_j^{(n)} = \sum_{1 \le i_1 < i_2 < \cdots < i_j \le n} x_{i_1} \cdots x_{i_j} \tag{19}$$

$$= x_n \left(\sum_{1 \le i_1 < i_2 < \cdots < i_{j-1} \le n-1} x_{i_1} \cdots x_{i_{j-1}} \right) + \sum_{1 \le i_1 < i_2 < \cdots < i_j \le n-1} x_{i_1} \cdots x_{i_j} = e_{j-1}^{(n-1)} x_n + e_j^{(n-1)} \tag{20}$$

Proposition 4.1.1. *Consider the following generating function:*

$$E(t) = \prod_{j=1}^{n} (1 + x_j t)$$

Then, if we denote $E_n(t)$, the same generating function truncated at the n^{th} product, then we have

$$E_n(t) = \prod_{j=1}^{n} (1 + x_j t) = \sum_{j=0}^{n} e_j^{(n)} t^j$$

Proof. We proceed by induction on n. Let $n = 1$. Then, the conclusion holds trivially. Now, for the induction step, let $n > 1$ and assume that the result holds for $n - 1$. Then, using the identity in line (20), together with the induction hypothesis, we have

$$\sum_{j=0}^{n} e_j^{(n)} t^j = \sum_{j=1}^{n} \left(e_{j-1}^{(n-1)} x_n + e_j^{(n-1)} \right) t^j = t x^n \prod_{j=1}^{n-1} (1 + t x_j) + \prod_{j=1}^{n-1} (1 + t x_j) \tag{21}$$

$$(1 + t x_n) \prod_{j=1}^{n-1} (1 + t x_j) = \prod_{j=1}^{n} (1 + t x_j) = E_n(t) \tag{22}$$

∎

Theorem 4.1.2 (Fundamental Theorem of Symmetric Polynomials)**.** *Λ_n is a polynomial ring in the n elementary symmetric polynomials. That is, every element $f \in \Lambda_n$ has a unique representation $\tilde{f}(e_1^{(n)}, \cdots, e_n^{(n)})$, where $\tilde{f} \in \mathbb{Z}[x_1, \cdots, x_n]$.*

Proof. We have to show that any symmetric polynomial can be expressed uniquely as a polynomial in the elementary symmetric functions $\{e_i\}_{i=1}^n$ with integer coefficients. We will show this by simultaneously inducting on the degree d of the symmetric polynomial $f \in \Lambda_n$ and on n. If $d = 0, 1$ or $n = 0, 1$, the result is trivial, hence the base case holds.

For the induction step, let $f \in \Lambda_n$ such that $deg(f) = d$ and assume the conclusion is true for all f' such that $deg(f') = d' < d$ and f' is a symmetric polynomial in at most $n' < n$ variables. Let $\pi : \mathbb{Z}[x_1, \cdots, x_n] \mapsto \mathbb{Z}[x_1, \cdots, x_{n-1}]$ denote the canonical evaluation map at $x_n = 0$. Now, we consider different cases:

Case 1 ($\pi(f) = 0$)**.** This implies that $x_n | f$, but f is a symmetric polynomial so that $x_i | f$ for all $i \in [n]$. In particular, $e_n | f$ and $deg(e_n | f) < d$. By our induction hypothesis, there exists a unique $h \in \mathbb{Z}[x_1, \cdots, x_n]$ such that $f = e_n h(e_1^{(n)}, \cdots, e_n^{(n)})$. So this case follows.

Case 2 ($\pi(f) \neq 0$)**.** In this case, $\pi(f)$ must be a polynomial of degree at most d in $n-1$ variables. By our induction hypothesis, we have that there is a unique $g \in \mathbb{Z}[x_1, \cdots, x_{n-1}]$ such that $g(e_1^{(n-1)}, \cdots, e_{n-1}^{(n-1)}) = \pi(f) = f(x_1, \cdots, x_{n-1}, 0)$. Now, recall the recursive identity presented in 20:

$$e_i^{(n)} = e_{i-1}^{(n-1)} x_n + e_i^{(n-1)} \quad \text{for all i} \in [n] \tag{23}$$

Hence, whenever $x_n = 0$, we have that $e_i^{(n)} = e_i^{(n-1)}$. This implies that $f - g(e_1^{(n)}, \cdots, e_{n-1}^{(n)})$ is symmetric, and since π is a ring isomorphism, $\pi(f - g(e_1^{(n)}, \cdots, e_{n-1}^{(n)})) = 0$. Thus, we are in **Case 1**, so that, by the same argument as above, there exists a unique $h \in \mathbb{Z}[x_1, \cdots, x_n]$ such that $f = e_n h(e_1^{(n)}, \cdots, e_n^{(n)}) - g(e_1^{(n)}, \cdots, e_{n-1}^{(n)})$. The result follows as required.

■

Essentially, 4.1.2 shows that the collection $\{e_j^{(n)}\}_{j=1}^n$ forms a basis for Λ_n. Now, we define yet another symmetric polynomial.

For $n \geq 1$ and $k \geq 0$ we denote by $h_j(x_1, \cdots, x_n)$ the *complete homogeneous symmetric polynomial of degree* j in n variables. Such is defined in terms of the monomial symmetric polynomial associated with the partition (j), which represents the tableau with a single row with j boxes:

$$h_j(x_1, \cdots, x_n) := m_{(j)}(x_1, \cdots, x_n) = \sum_{1 \leq i_1 \leq i_2 \leq \cdots \leq i_j \leq n} x_{i_1} \cdots x_{i_j} \tag{24}$$

Right away, we see that the following identity holds:

$$h_j(x_1, \cdots, x_n) = \sum_{|\lambda| = j} m_\lambda(x_1, \cdots, x_n) \tag{25}$$

which then shows that indeed, $h_j^{(n)}$ is a symmetric polynomial

Proposition 4.1.3. *Consider the following generating function*

$$H(t) := \prod_{j=1}^{\infty} \frac{1}{1 - x_j t}.$$

Then, if we denote $H_n(t)$, the same generating function truncated at the n^{th} product, then we have

$$H_n(t) = \prod_{j=1}^{n} \frac{1}{1 - x_j t} = \sum_{j=0}^{n} h_j^{(n)} t^j$$

Proof. The Geometric Series formula tells us that $\frac{1}{1-tx_j} = \sum_{k=0}^{\infty} x_j^k t^k$. Fix n. Define $\bar{h}_j$ to be the coefficent associated with the formal power $H_n(t)$, *i.e*

$$H_n(t) = \prod_{j=1}^{n} \sum_{k=0}^{\infty} x_j^k t^k = \sum_{j=0}^{n} \bar{h}_j t^j$$

We can already see that $\bar{h}_j$ is going to be a sum over the monomials in the variables x_i. How do we gather all the monomials on the RHS of this expression so that we can fix the exponent of t? Fix $0 \leq j \leq n$. Say that we observe monomials on the LHS to be $x_{j_1}^{k_1} \cdots x_{j_m}^{k_m}$. Note that, for any $1 \leq r \leq k$, the exponent k_r is equal to the exponent of t on the product in the RHS, it must be that this monomial is accompanying $j = k_1 + \cdots + k_m$ on the LHS. Conversely, every monomial appearing in this sum has this degree k. Thus, the coefficent of t^j, *i.e* $\bar{h}_j$, is precisely the sum of all degree j monomials in x_i's.

Thus, permuting the indexes to collect terms that lie in the same orbit, we have that

$$\bar{h}_j = \sum_{1 \leq i_1 \leq i_2 \leq \cdots i_j} x_{i_1} x_{i_2} \cdots x_{i_j} = h_j^{(n)}$$

■

We can now combine 4.1.3 and 4.1.1 to get the an interesting relationship between them. For all n natural, we have

$$H_n(t) E_n(-t) = 1$$

. In particular, if we expand this identity, we have:

$$1 = \sum_{k=0}^{n} h_k^{(n)} t^k \sum_{j=0}^{n} e_j^{(n)} (-t) t^j = \sum_{k=0}^{n} \left(\sum_{j=0}^{k} (-1)^j e_j^{(n)} h_{k-j}^{(n)} \right) t^k \tag{26}$$

$$\implies \quad \text{For all } k, n \geq 1, \quad \sum_{j=0}^{k} (-1)^j e_j^{(n)} h_{k-j}^{(n)} = 0 \tag{27}$$

$$\implies h_k^{(n)} = \sum_{j=1}^{k} (-1)^{j+1} e_j^{(n)} h_{k-j}^{(n)} \quad \textbf{and} \quad e_k^{(n)} = \sum_{j=1}^{k} (-1)^{j+1} h_j^{(n)} e_{k-j}^{(n)} \tag{28}$$

With this identity in our toolbox, we can now prove that $\{h_j^{(n)}\}_{j=0}^{n}$ forms a basis for Λ_n, just as in 4.1.2

Theorem 4.1.4. Λ_n *is a polynomial ring in the first n degree complete homogeneous polynomials in n variables* $\{h_j^{(n)}\}_{j=1}^n$. *That is, every element* $f \in \Lambda_n$ *has a unique representation* $\tilde{f}(h_1^{(n)}, \cdots, h_n^{(n)})$, *where* $\tilde{f} \in \mathbb{Z}[x_1, \cdots, x_n]$.

Proof. Since this result in true for $\{e_j^{(n)}\}_{j=1}^n$ (4.1.2), it suffices to create an endomorphism φ such that $\varphi(e_j) = h_j$. In fact we define φ in this suggestive manner, *i.e* let $\varphi : \Lambda_n \mapsto \Lambda_n$ such that $\varphi(e_j^{(n)}) = h_j^{(n)}$, for all $0 \leq j \leq n$. This completely determines φ since $\{e_j^{(n)}\}_{j=1}^n$ forms a basis for Λ_n. Now, it suffices to show that φ in invertible. We will show an even stronger statement. Namely, that φ is its own inverse (an involution).

We proceed by induction on j. Consider the base case, $j = 0$. Then, $h_0^{(n)} = 1 = e_0^{(n)}$. Now, let $j > 0$ and assume that the result is true for all $k < j$. Then, using the recursive identity 20, 28, and the inductive hypothesis, we get

$$\varphi^2(e_j^{(n)}) = \varphi(h_j^{(n)}) = \varphi(\sum_{k=1}^{j}(-1)^{k+1}e_k^{(n)}h_{j-k}^{(n)}) = \sum_{k=1}^{j}(-1)^{k+1}h_k^{(n)}e_{j-k}^{(n)} = e_j^{(n)}$$

Indeed, φ is an involution, as required. ∎

Corollary 4.1.5. *The collection of monomial symmetric polynomials* $\{m_\lambda^{(n)}\}_{l(\lambda)\leq n}$ *is a* $\mathbb{Z}$ *bases for* Λ_n

Proof. This follows from 4.1.4 and the identity in 25 ∎

4.2 Schur Polynomials

We will now introduce another symmetric polynomial which is going to be an important object of our study throughout the remaining of this report: **Schur polynomials**.

Just as with the theory of symmetric polynomials, Schur polynomials can be defined in many ways. Schur polynomials were originally introduced by Isaii Schur, whose definition of such involves a ratio of determinants. We will present this algebraic definition for consistency.

Let λ be an integer partition such that. Then, for define for $n \geq \lambda$, we define the matrix

$$A_\lambda(x_1, \cdots, x_n) := \begin{bmatrix} x_1^{\lambda_1} & x_2^{\lambda_1} & \dots & x_n^{\lambda_1} \\ x_1^{\lambda_2} & x_2^{\lambda_2} & \dots & x_n^{\lambda_2} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{\lambda_n} & x_2^{\lambda_n} & \dots & x_n^{\lambda_n} \end{bmatrix}$$

Then, we define the Schur polynomial $s_\lambda(x_1, \cdots, x_n) \in \Lambda_n$ to be

$$s_\lambda(x_1, \cdots, x_n) := \frac{det(A_{\lambda+\delta^{(n)}}(x_1, \cdots, x_n))}{det(A_{\delta^{(n)}}(x_1, \cdots, x_n))} \tag{29}$$

For a remainder on the $\delta^{(n)}$ partition, refer to 2.1.

At first glance, this is something hard to chew on. It is a ratio of determinants, so one has to show that this is well defined. Moreover, it is not even clear why exactly these polynomials are symmetric. We will not attempt to move forward with this definition of the schur polynomials. However, keep in mind that it can be done.

We will stir away from this algebraic approach and give a definition which is directly related to the study of tableaux. Of course, the two definitions are consistent, and, this fact alone, already illustrates how the combinatorics of tableaux aids in the study of these polynomials.

If T is a tableau of shape λ with entries in $[n]$, denote by x^T the product of all x_i where $i \in [n]$ and with multiplicity that of the entries of T. That is,

$$x^T := \prod_{i=0}^{n} x_i^{\alpha_i}$$

where α_i denotes the multiplicity of entry i in T. The Schur polynomial $s_\lambda(x_1, \cdots, x_n) \in \Lambda_n$ is defined to be

$$s_\lambda(x_1, \cdots, x_n) := \sum_{T \text{ on } \lambda} x^T \tag{30}$$

The latter generalizes naturally for skew tableaux T of shape λ/ρ, with entries in $[n]$:

$$s_{\lambda/\rho}(x_1, \cdots, x_n) := \sum_{T \text{ on } \lambda/\rho} x^T \tag{31}$$

Proposition 4.2.1. *The Schur Function is a symmetric function.*

Proof. Fix a partition λ, $\in \mathbb{N}$, and $\sigma \in S_n$. Consider $s_\lambda(x_{\sigma(1)}, \cdots, x_{\sigma(n)})$. Then note that the order of the inputs $\{x_1, \cdots, x_n\}$ determine the x^T insofar as they tell how are we taking into account the multiplicity of the entries of T. Hence, by 3.1.8, the result follows. ■

Now, the next proposition relates the schur polynomials to the homogeneous symmetric polynomials:

Proposition 4.2.2. *Let λ be a partition with entries in [n]. Then,*

$$s_\lambda(x_1, \cdots, x_n)h_j(x_1, \cdots, x_n) = \sum_{\mu} s_\mu(x_1, \cdots, x_n) \tag{32}$$

Where μ is summed over all partitions obtained from λ by adding j boxes, none of which lie in the same column.

Proof. Note that , using 24

$$s_\lambda(x_1, \cdots, x_n)h_j(x_1, \cdots, x_n) = s_\lambda(x_1, \cdots, x_n)m_{(j)}(x_1, \cdots, x_n) \tag{33}$$

$$= \sum_{T \text{ on } \lambda} x^T \sum_{1 \le i_1 \le i_2 \le \cdots \le i_j \le n} x_{i_1} \cdots x_{i_j} = \sum_{T \text{ on } \lambda} x^T \sum_{J \text{ on } (j)} x^J \tag{34}$$

Firstly, recall that the set of tableau with entries in $[n]$, $\mathcal{T}_n$ is monoid under the operation of concatenation of words $(\cdot)$ (2.7.1). Thus, since every monoid has an associated ring, say R_n, one may think of the map $T \mapsto x^T$ as a ring homomorphism between the R_n and Λ_n. In particular, under this homomorphism, h_j is the image of the sum of all tableaux with j boxes in the first row. Thus, computing one term in the RHS of 24, amounts to evaluating the image (under said homomorphism) of a tableau given by the word $w(T)w(J)$, where T with shape λ and entries in $[n]$ and J represents a tableau with j boxes in the first row. But, remember that $w(T)\cdot w(J) \equiv w(T\cdot J)$ and the tableau associated with $w(T\cdot J)$ is given by row inserting the word of J from left to right into T. Thus, indeed since $w(J) = w_1 \cdots w_j$ increases from left to right, 2.5.3 gives us that $w(T\cdot J)$ is going to be associated with a new tableau on the shape μ with j new boxes, none in the same column. ■

We have an analogous proposition for elementary symmetric polynomials:

Proposition 4.2.3. *Let λ be a partition with entries in $[n]$. Then,*

$$s_\lambda(x_1, \cdots, x_n)e_j(x_1, \cdots, x_n) = \sum_{\rho} s_\rho(x_1, \cdots, x_n) \tag{35}$$

Where ρ is summed over all partitions obtained from λ by adding j boxes, none of which lie in the same row

Proof. Again, we have

$$s_\lambda(x_1, \cdots, x_n)e_j(x_1, \cdots, x_n) = s_\lambda(x_1, \cdots, x_n)m_{(1)^j}(x_1, \cdots, x_n) \tag{36}$$

$$= \sum_{T\,\text{on}\,\lambda} x^T \sum_{1\le i_1<i_2<\cdots<i_j\le n} x_{i_1}\cdots x_{i_j} = \sum_{T\,\text{on}\,\lambda} x^T \sum_{J\,\text{on}\,(1^j)} x^J \tag{37}$$

Again, the exact same argument applies, the only difference is that $w(J)$ is now strictly increasing. But again, the row bumping lemma 2.5.3, yields the necessary result about the shape of $w(T\cdot J)$. The result follows. ■

Now, we will show how the RSK correspondence 3.1.3 can be used to give a proof of the Cauchy Identity:

Theorem 4.2.4 (Cauchy Identity)**.** *The following identity holds:*

$$\prod_{i=1}^{n}\prod_{j=1}^{m} \frac{1}{1 - x_i y_j} = \sum_{\lambda\in\Upsilon} s_\lambda(x_1, \cdots, x_n)s_\lambda(y_1, \cdots, y_m) \tag{38}$$

Proof. As commented right after 3.1.3, RSK gives a correspondence between pairs of same shape tableaux (P, Q), where P is a tableau with entries in $[n]$ and Q is a tableau with entries in $[m]$, and $n \times m$ matrices A with non-negative integer entries. For any $m \times n$ matrix A with non-negative integer entries, define $(xy)^A := \prod_{i=1}^{n}\prod_{j=1}^{m}(x_i y_j)^{a_{ij}}$.

Let A be an $m \times n$ matrix associated with the pair (P, Q) of same shape tableaux (via RSK), where P, Q have entries in $[n], [m]$ respectively. The construction of this associated matrix A tells us that its entry $a_{i,j}$ stands for the multiplicity of the entry j ocurring in P and the multiplicity of of the entry i occurring in Q. Hence, we have:

$$\sum_{\lambda\in\Upsilon} s_\lambda(x_1, \cdots, x_n)s_\lambda(y_1, \cdots, y_m) = \sum_{P,Q\,\text{on}\,\lambda} x^P y^Q \tag{39}$$

$$= \sum_{\substack{A,\, m\times n \\ \text{nonegative entries}}} (xy)^A = \sum_{\substack{A,\, m\times n \\ \text{nonegative entries}}} \prod_{i=1}^{n}\prod_{j=1}^{m} (x_i y_j)^{a_{ij}} \tag{40}$$

But, recall that $(1-x_iy_j)^{-1} = (1+x_iy_j+x_i^2y_j^2+\cdots)$. Hence, the RHS amounts to choosing one term from the coefficent of the formal power series related to the geometric series $(1-x_iy_j)^{-1}$. Indeed, we have

$$\sum_{\lambda\in\Upsilon} s_\lambda(x_1,\cdots,x_n)s_\lambda(y_1,\cdots,y_m) = \sum_{\substack{A,\, m\times n \\ \text{nonegative entries}}} \prod_{i=1}^{n}\prod_{j=1}^{m}(x_iy_j)^{a_{ij}} = \prod_{i=1}^{n}\prod_{j=1}^{m}\frac{1}{1-x_iy_j} \tag{41}$$

■

Proposition 4.2.5. *The following identity for Schur polynomials hold:*

$$\prod_{i=1}^{m}(1-x_i)^{-1}\prod_{1\le i<j\le m}(1-x_ix_j)^{-1} = \sum_{\lambda\in\Upsilon} s_\lambda(x_1,\cdots,x_m) \tag{42}$$

Proof. Again, we use the geometric series identity $(1-x_i)^{-1} = (1+x_i+x_i^2+\cdots)$. Hence, if we denote $\mathcal{A}_{[m\times n]}$, to be all the possible $m\times m$ **symmetric** matrices with non-negative integer entries, we have:

$$\prod_{i=1}^{m}(1-x_i)^{-1}\prod_{1\le i<j\le m}(1-x_ix_j)^{-1} = \sum_{A\in\mathcal{A}_{[m\times m]}}\left(\prod_{i=1}^{m}x_i^{a_{ii}}\prod_{i<j}(x_ix_j)^{a_{ij}}\right) \tag{43}$$

Now, on the RHS of line 42, we have the summation of all possible monomials of the form x^P, where P is a tableau on λ with entries in $[m]$. Thus, it suffices to apply the correspondence between tableaux P and symmetric matrices A, given in 3.1.6. The result follows. ■

.

Proposition 4.2.6. *The collection of schur polynomials $\{s_\lambda^{(n)}\}_{l(\lambda)\le n}$ is a $\mathbb{Z}$ bases for Λ_n*

Proof. It suffices to show that, for each $n\ge j\ge 0$, we have $e_j^{(n)} = s_{(1^j)}^{(n)}$ and then apply 4.1.2. This follows easily:

$$e_j^{(n)} = \sum_{1\le i_1<i_2<\cdots<i_j\le n} x_{i_1}\cdots x_{i_j} = \sum_{J\text{ on }(1^j)} x^J = s_{(1^j)}^{(n)} \tag{44}$$

■

Just like this last identity we just showed, we also have that, for each $n\ge j\ge 0$,

$$h_j^{(n)} = \sum_{1\le i_1\le i_2\le\cdots\le i_j\le n} x_{i_1}\cdots x_{i_j} = \sum_{J\text{ on }(j)} x^J = s_{(j)}^{(n)} \tag{45}$$

In fact, these two identities, together with the map φ described in 4.1.4, can be used to prove 4.2.2 and 4.2.3 directly. Along these lines, applying φ to 4.2.2 and 4.2.3 implies that $\varphi(\lambda) = \varphi(\lambda')$, which can also be an useful identity.

Before we go any further in our study of symmetric polynomials, we will discuss their generalizations, symmetric functions, which are essentially symmetric polynomials in infinitely many variables. We will make that precise in the following section.

4.3 Ring of Symmetric Functions

Thus far, we have concerned ourselves only with symmetric polynomials. We now introduce a bit more complex object, which can be viewed as an generalization of symmetric polynomials.

This section could look unmotivated to a neophyte, thus one should not be alarmed upon not being able to comprehend it entirely. However, there are key ideas here that are worth mentioning. What follows is best motivated by the fact that it is useful to develop a theory that allows us to handle symmetric polynomials in infinitely many variables.

At the same time, this theory should be consistent with the identities and properties developed already for symmetric polynomials in finitely many variables. The latter is usually achieved by requiring that, when setting all but finitely many variables to zero, these functions are identical to polynomials.

Let Λ_n denote the space of symmetric polynomials in n variables over $\mathbb{Z}$. We will construct the ring of symmetric functions using the algebraic technique of the inverse limit (for reference on this particular technique, see [10]).

Let Λ_n^k be the subset of Λ_n consisting of the degree k symmetric polynomials in n variables. Then, we have a family $\{\Lambda_n^k\}_{n\geq 0}$ of sets indexed by the natural numbers including 0. For each $i \leq j \in \mathbb{N}$, we have the canonical homomorphism $\pi_{ij}^k : \Lambda_j^k \mapsto \Lambda_i^k$, via

$$\pi_{ij}^k(f(x_1, \cdots, x_i, x_{i+1}, \cdots, x_j) \mapsto f(x_1, \cdots, x_i, 0, \cdots, 0) \tag{46}$$

Clearly, we have that $\pi_{ii}^k(f) = f$ for all $f \in \Lambda_i^k$, and for all $i \in \mathbb{N}$. Moreover, for all $f \in \Lambda_r^k$, and for all $i \leq j \leq r$

$$\pi_{ij}^k \circ \pi_{jr}^k(f) = f(x_1, \cdots, x_i) = \pi_{ir}^k(f) \tag{47}$$

Note that these maps are an isomorphism whenever $j \leq i \leq n$ (because these polynomials are symmetric). Now, we are ready to define the inverse limit of the collection $\{\Lambda_n^k\}_{n\geq 0}$:

$$\Lambda^k := \varprojlim_n \Lambda_n^k := \{\vec{u} \in \prod_{n\geq 0} \Lambda_n^k \,|\, \pi_{ij}^k(u_j) = u_i \text{ for all } i \leq j\} \tag{48}$$

We call Λ^k the symmetric functions of degree k. Formally, its elements are sequences $\{f_n\}$, where $f_n \in \Lambda_n^k$ such that its j^{th} element f_j agrees with its i^{th} element whenever we evaluate the variables $x_{i+1}, \cdots, x_j$ at 0. These are precisely the functions whose component wise sequence satisfy our initial intuition for the what **should** be a symmetric function. Thus, we may now rigorously think of the extension of a symmetric polynomial of degree k to infinitely many variables.

Furthermore, Λ_k comes endowed with the map $\pi_n^k : \Lambda^k \mapsto \Lambda_n^k$, via $f \mapsto f_n$, its n^{th} component. Again, because these functions are symmetric, these maps are isomorphisms whenever $n \geq k$. In particular, since the collection of monomial symmetric polynomials $\{m_\lambda^{(n)}\}_{l(\lambda)\leq n}$ is a $\mathbb{Z}$ bases for $\Lambda_n \supset \Lambda_n^k$, $\{m_\lambda\}_{\lambda\vdash k}$ is a basis for Λ^k.

We are now ready to define our main object of study in this section **ring of symmetric functions**. Define

$$\Lambda := \bigoplus_{k\geq 0} \Lambda^k \tag{49}$$

Where, $\oplus$ denotes the direct sum of vectors spaces. Also, define the surjective ring homomorphisms $\pi_n : \Lambda \mapsto \Lambda_n$ via:

$$\pi_n := \bigoplus_{k\geq 0} \pi_n^k \tag{50}$$

Recall that π_n^k is an isomorphism whenever $n \geq k$. In particular, if we then restrict π_n to the first the direct product over $k \in \{0, \cdots, n\}$, then π_n becomes an isomorphism into Λ_n^k. Effectively, what this is saying is that the space of symmetric functions of degrees of degree at most n is isomorphic to the space of symmetric polynomials in n variables of degree at most n.

This characterization of this map allows us to any proposition that we have proven thus far for arbitrary symmetric polynomials. Suppose that we have any identity which involves symmetric polynomials of degree at most n in n variables. Then, by the preceding discussion, this identity will hold for the space of symmetric functions with degree at most n! This discussion produces a immediate proof for the following very useful proposition:

Proposition 4.3.1. *If, for all n, an identity holds for homogeneous symmetric polynomials in n variables and of degree at most $k \leq n$. Then, the identity is valid for Λ.*

For symmetric functions $f \in \Lambda$, we omit the entry of variables and usually write it as $f(x)$ to mean $f(x_1, x_2, \cdots)$. Sometimes, we will even completely drop (x) and write f. One example of how the polynomials we have thus far studied generalize is the schur function s_λ. Remember that $s_\lambda(x_1, \cdots, x_n) = \sum_{T \text{ on } \lambda} x^T$, where T had entries in $[n]$. Now, the schur function $s_\lambda = \sum_{T \text{ on } \lambda} x^T$, but now T has arbitrary non-negative integer entries. Thus, a much larger class of tableaux T are considered for the schur function (in fact, an infinitely larger class).

Corollary 4.3.2. *Both collections of symmetric functions $\{m_\lambda\}_{\lambda \in \Upsilon}$, $\{s_\lambda\}_{\lambda \in \Upsilon}$, form a $\mathbb{Z}$ bases for Λ.*

Proof. Combine 4.1.4 and 4.2.6 with 4.3.1 above ■

Theorem 4.3.3. *The formal power series regarding the symmetric functions $h_i(x)$ and $e_i(x)$ admit the following identities:*

$$\sum_{n \geq 0} h_i(x) t^i = \prod_{i \geq 1} \frac{1}{1 - x_i t} \quad \textit{and} \quad \sum_{n \geq 0} e_i(x) t^i = \prod_{i \geq 1} (1 + x_i t) \tag{51}$$

Proof. This is virtually the same proof as exhibit in 4.1.3 and 4.1.1, but now we are able to make sense of h_n, e_n as symmetric functions, *i.e* a symmetric polynomials in infinitely many variables. Thus, one does not have to truncate the product at a finite step n. ■

We now introduce a generalization of the elementary symmetric functions e_j and the homogeneous symmetric functions h_j. Let $\lambda \in \Upsilon$. Then, we define

$$e_\lambda = e_{\lambda_1} e_{\lambda_2} \cdots \tag{52}$$

$$h_\lambda = h_{\lambda_1} h_{\lambda_2} \cdots \tag{53}$$

Recall that a partition $\lambda = (\lambda_1, \lambda_2, \cdots) \in \Upsilon$ is such that $\lambda_1 \geq \lambda_2 \geq \cdots$ and $\lambda_i = 0$ for all but finitely many i.

Theorem 4.3.4. *The following collections form a $\mathbb{Z}$ bases for Λ. The symmetric functions $\{e_\lambda\}_{\lambda \in \Upsilon}$ (respectively, $\{h_\lambda\}_{\lambda \in \Upsilon}$), is a $\mathbb{Z}$ basis for Λ.*

Proof. Recall that the collections $\{e_j^{(n)}\}_{j=1}^n$, $\{h_j^{(n)}\}_{j=1}^n$, $\{s_\lambda^{(n)}\}_{\lambda\in\Upsilon}$ form $\mathbb{Z}$ bases for Λ_n. Using, the identities 44, 45, we see that $\{e_\lambda^{(n)} : \lambda \text{ has at most n columns}\}$, and $\{h_\lambda^{(n)} : \lambda \text{ has at most n rows}\}$ are $\mathbb{Z}$ bases for Λ_n. Apply 4.3.1. ∎

Proposition 4.3.5. *The following identity holds in the ring of symmetric functions:*

$$\sum_{\lambda\in\Upsilon} s_\lambda(x)s_\lambda(y) = \sum_{\lambda\in\Upsilon} h_\lambda(x)m_\lambda(y) \tag{54}$$

Proof. Using 4.3.3 and the cauchy identity 4.2.4 (for the ring of symmetric functions), we see that

$$\sum_{\lambda\in\Upsilon} s_\lambda(x)s_\lambda(y) = \prod_{i=1}^{\infty}\prod_{j=1}^{\infty}\frac{1}{1-x_iy_j} = \prod_{j=1}^{\infty}\left(\sum_{k\geq 0} h_k(x)y_j^k\right) = \sum_{\lambda\in\Upsilon} h_\lambda(x)m_\lambda(y) \tag{55}$$

∎

4.4 Hall inner product

As discussed in the previous section, the collections of schur polynomials $\{s_\lambda\}_{\lambda\in\Upsilon}$ form a $\mathbb{Z}$ basis for Λ, the ring of symmetric functions. But so does the other collections $\{e_\lambda\}_{\lambda\in\Upsilon}$, $\{m_\lambda\}_{\lambda\in\Upsilon}$, $\{h_\lambda\}_{\lambda\in\Upsilon}$. In the beginning of our discussion of symmetric polynomials, we have mentioned that our main object of study would be the schur polynomials, but such a focus has not yet been justified.

We will now see how the collection of schur polynomials $\{s_\lambda\}_{\lambda\in\Upsilon}$, form a orthonormal basis for Λ under the hall inner product. We create the hall inner product so that the basis $\{h_\lambda\}_{\lambda\in\Upsilon}$ and $\{m_\lambda\}_{\lambda\in\Upsilon}$ are duals to one another.

Definition 4.4.1 (Hall Inner Product)**.** *Define $\langle\cdot,\cdot\rangle$ to be the bilinear map from $\Lambda \mapsto \mathbb{R}$ such that it acts on the basis elements $\{m_\mu\}_{\mu\in\Upsilon}$, $\{h_\lambda\}_{\lambda\in\Upsilon}$ as follows:*

$$\langle h_\lambda, m_\mu\rangle = \delta_{\lambda\mu} := \begin{cases} 1 & \text{if } \lambda = \mu \\ 0 & \text{if } \lambda \neq \mu \end{cases} \tag{56}$$

This definition completely defines the inner-product on Λ, as each symmetric function $f \in \Lambda$ has a unique representation in terms of $\{m_\mu\}_{\mu\in\Upsilon}$ and $\{h_\lambda\}_{\lambda\in\Upsilon}$. Now, to conclude that this is indeed an inner product, we have to check that it is positive definite (i), symmetric (ii), and linear in the first argument (iii). Note that (iii) follows right away from the very requirement of its definition, *i.e* that the map is bilinear. To check (i) and (ii) we will prove a proposition first:

Proposition 4.4.2. *Let $\lambda, \mu \in \Lambda$. Then $\langle s_\lambda, s_\mu\rangle = \delta_{\lambda\mu}$.*

Proof. Write $s_\lambda = \sum_{\rho\in\Upsilon} a_{\lambda\rho}h_\rho$ and $s_\mu = \sum_{\rho\in\Upsilon} b_{\mu\rho}m_\rho$. Then, we have

$$\langle s_\lambda, s_\mu\rangle = \langle\sum_{\rho\in\Upsilon} a_{\lambda\rho}h_\rho, \sum_{\nu\in\Upsilon} b_{\mu\nu}m_\nu\rangle = \sum_{\rho\in\Upsilon} a_{\lambda\rho}\left(\langle h_\rho, \sum_{\nu\in\Upsilon} b_{\mu\nu}m_\nu\rangle\right) = \sum_{\rho\in\Upsilon} a_{\lambda\rho}\left(\sum_{\nu\in\Upsilon} b_{\mu\nu}\langle h_\rho, m_\nu\rangle\right) \tag{57}$$

$$= \sum_{\rho\in\Upsilon} a_{\lambda\rho}\left(\sum_{\nu\in\Upsilon} b_{\mu\nu}\delta_{\rho\nu}\right) = \sum_{\rho\in\Upsilon} a_{\lambda\rho}b_{\mu\rho} \tag{58}$$

Now, using 4.3.5, we also see that

$$\sum_{\rho\in\Upsilon} h_\rho(x)m_\rho(y) = \sum_{\rho\in\Upsilon} s_\rho(x)s_\rho(y) = \sum_{\rho\in\Upsilon}\left[\left(\sum_{\lambda\in\Upsilon} a_{\rho\lambda}h_\lambda(x)\right)\left(\sum_{\mu\in\Upsilon} b_{\rho\mu}m_\mu(y)\right)\right] \tag{59}$$

$$\implies \sum_{\rho\in\Upsilon} a_{\rho\lambda}b_{\rho\mu} = 1 \text{ iff } \mu=\lambda \tag{60}$$

Now putting lines 58, 58 together, we get the desired equality. ■

Proposition 4.4.3. *The hall product defined above is an inner product.*

Proof. As argued, we check that it is positive definite and symmetric. Let $u, v \in \Lambda$. Then, using the fact that $\{s_\mu\}_{\mu\in\Upsilon}$ is a basis for Λ, write $u = \sum_\mu a_\mu s_\mu$ and $v = \sum_\nu b_\nu s_\nu$. Then, the following chain of equalities show the desired result:

$$\langle u, v\rangle = \sum_{\mu\in\Upsilon} a_\mu\left(\sum_{\nu\in\Upsilon} b_\nu\langle s_\mu, s_\nu\rangle\right) = \sum_{\mu\in\Upsilon} a_\mu\left(\sum_{\nu\in\Upsilon} b_\nu\delta_{\mu\nu}\right) = \sum_{\mu\in\Upsilon} a_\mu\left(\sum_{\nu\in\Upsilon} b_\nu\delta_{\nu\mu}\right) \tag{61}$$

$$= \sum_{\nu\in\Upsilon} b_\nu\left(\sum_{\mu\in\Upsilon} a_\mu\langle s_\nu, s_\mu\rangle\right) = \langle v, u\rangle \tag{62}$$

This shows that the product $\rangle\langle$ is symmetric as required. Now, we will show that it is positive definite. Consider u as above. Then,

$$\langle u, u\rangle = \sum_{\mu\in\Upsilon} a_\mu\left(\sum_{\nu\in\Upsilon} a_\mu\langle s_\mu, s_\nu\rangle\right) = \sum_{\mu\in\Upsilon} a_\mu^2 \geq 0 \tag{63}$$

Furthermore, $\langle u, u\rangle = 0$ iff $\{a_\mu\} = 0$, which happens iff $u = 0$. Hence, the hall product is an inner product on Λ as required ■

As promised, we have the proposition, whose proof is immediate from the above two propositions.

Proposition 4.4.4. $\{s_\lambda\}_{\lambda\in\Upsilon}$ *is a orthonormal basis for* Λ *under the hall inner product.*

4.5 Littlewood-Richardson Coefficients

In the last section, we showed that $\{s_\lambda\}_{\lambda\in\Upsilon}$ forms orthonormal $\mathbb{Z}$ basis for Λ. Hence, any element $u \in \Lambda$ can be uniquely determined by its hall inner product with any schur polynomial, *i.e* $\langle u, s_\lambda\rangle$. But how exactly can we go about multiplying schur functions?

Well, since Λ is a ring, given two schur functions s_λ, s_μ on Λ, then $s_\lambda s_\mu \in \Lambda$ and hence, it can be expressed as a linear combination of $\{s_\nu\}_{\nu\in\Upsilon}$. Denote these coefficients by $\{c^\nu_{\lambda\mu}\}_{\nu\in\Upsilon}$.

$$s_\lambda s_\mu = \sum_{\nu\in\Upsilon} c^\nu_{\lambda\mu} s_\nu \tag{64}$$

These coefficients are of extreme importance in the understanding of schur polynomials. So much so that they receive a special name: The Littlewood-Richardson Coefficients.

There are many ways one can go about arriving at them. For instance, we can just declare them to satisfy 64 above. This is completely fine, since we already know that the schur functions are a basis for the space. However, this gives poor insight what they actually reveal about λ and μ.

For the purposes of highlighting the combinatorics of the tableaux, we will pretend that defining these directly is not possible. Instead, starting with partitions λ, μ, and ν, we will construct $c^{\nu}_{\lambda\mu}$ using our knowledge of the combinatorics of tableaux. Only after that, we will show said coefficients satisfy 64.

To start off, we will need the following lemma, which ties the RSK correspondence with skew tableaux.

Lemma 4.5.1. *Suppose the lexicographic array* $w = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$ *is associated with the tableaux* (P, Q) *via the* RSK *correspondence. Let* T *be any tableau and let*

$$T' = (\cdots((T \leftarrow v_1) \leftarrow v_2) \cdots \leftarrow v_{r-1}) \leftarrow v_r$$

and place $u_1, \cdots, u_r$ *successively in the new boxes. Then, the entries* $u_1, \cdots, u_r$ *form a skew tableau* S *with shape* T'/T *and* $Rect(S) = Q$.

Proof. In the interest of time and conciseness, we will not produce a proof of this fact, but one can be found in [1], page 50. ■

Now, fix three partitions $\lambda \vdash n, \mu \vdash m, \nu \vdash r$. We want to know how many ways can we choose tableau T on λ and U on μ such that $rect(T * U) = T \cdot U = V$. Obviously, this wouldn't be possible if either of $r = n + m$ or $\lambda \subset \nu$ failed. By our discussion of $rect(T * U)$ presented in **??**, this is equivalent to ask the number of skew tableaux on the shape $\lambda * \mu$ whose rectification is R. Our aim with this discussion is to show that the number of ways to do the latter is the same as the number of skew tableaux on the shape ν/λ such that its rectification is μ. To make this precise, we introduce the following notation.

For any tableau U_0 on μ and V_0 on ν, we set

$$\mathcal{S}(\nu/\lambda, U_0) = \{\text{skew tableaux } S \text{ on the shape } \nu/\lambda \mid Rect(S) = U_0\} \tag{65}$$

$$\mathcal{T}(\lambda, \mu, V_0) = \{\text{pairs of tableaux (T,U)} \mid T \textit{ on } \lambda, U \textit{ on } \mu, \textit{ and } T \cdot U = V_0\} \tag{66}$$

Proposition 4.5.2. *Fix* λ, μ, *and* ν *partitions. Then, for any tableau* U_0 *on* μ *and* V_0 *on* ν *Then, we have*

$$|\mathcal{S}(\nu/\lambda, U_0)| = |\mathcal{T}(\lambda, \mu, V_0)| \tag{67}$$

Proof. It suffices to construct a bijective correspondence between the two sets. We will show how to go from $(T, U) \in \mathcal{T}(\lambda, \mu, V_0)$ to a unique element $S \in \mathcal{S}(\nu/\lambda, U_0)$. Let $(T, U) \in \mathcal{T}(\lambda, \mu, V_0)$ and consider the lexicographic array $w = \left(\begin{smallmatrix} v_1 & v_2 & \cdots & v_r \\ u_1 & u_2 & \cdots & u_r \end{smallmatrix}\right)$ corresponding to it under RSK 3.1.3. Let

$$T' = (\cdots((T \leftarrow v_1) \leftarrow v_2) \cdots \leftarrow v_{r-1}) \leftarrow v_r$$

and, let S as in lemma 4.5.1. Then, by the lemma, S has shape that of $T'/T = V_0/T$, which is precisely ν/λ, because $V_0 = T' = T \cdot U$. Since the two rowed array completely determines S, and is unique for each pair (U, U_0), we see that this is a bijection. ■

What is remarkable about 4.5.2 is how the sizes of $\mathcal{S}(\nu/\lambda, U_0)$ and $\mathcal{T}(\lambda, \mu, V_0)$ are not dependent on our choice of tableaux U_0 and V_0, as long as these tableaux are on the appropriate shape, namely λ, μ and ν.

Finally, we now arrive at the Littlewood-Richardson coefficient $c^{\nu}_{\lambda\mu}$, where

$$c^{\nu}_{\lambda\mu} := |\mathcal{S}(\nu/\lambda, U_0)| = |\mathcal{T}(\lambda, \mu, V_0)| \tag{68}$$

With this in mind, we are able to now prove 64.

Proposition 4.5.3. *The following identity holds in the ring of symmetric functions:*

$$s_\lambda s_\mu = \sum_{\nu \in \Upsilon} c^{\nu}_{\lambda\mu} s_\nu$$

Proof. Start by looking at the LHS of the equation. The number of times a monomial x^V, for some tableau V on $\nu \vdash r$, is precisely the number of times you can find monomials x^T, x^U, where T is on $\lambda \vdash n$, U is on $\mu \vdash m$ and $r = n + m$. How many times does that actually happen? By 4.5.2, exactly $c^{\nu}_{\lambda\mu}$ many times. But this value is independent of the entries in T, U so that we can gather terms depending only on ν. Summing all possible choices of ν we have the desired equality. ■

Now, we can obtain an alternative representation of the Littlewood-Richardson coefficients via skew tableaux:

Proposition 4.5.4. *The following identity holds in the ring of symmetric functions:*

$$s_{\nu/\lambda} = \sum_{\mu \in \Upsilon} c^{\nu}_{\lambda\mu} s_\mu \tag{69}$$

Proof. Recall that $s_{\nu/\lambda} = \sum_{\text{S on } \nu/\lambda} x^S$. Now, look at the RHS of what we want to prove. Fix some shape μ. Then, for some U on μ, x^U occur precisely $c^{\nu}_{\lambda\mu}$ on the RHS. Again, by 4.5.2, this is also the number of skew tableau S of shape ν/λ such that its rectification is U. Thus, since all such S will have the same content as in U, then we gather the terms x^S on the LHS and the inequality follows as required. ■

Remark. *Even though the former two identities are proven for the space symmetric functions, they do hold for the finitely variable schur polynomials as expected. One just has to restrict the sum over tableaux, limiting its entries accordingly. That is, for Λ_n, we have:*

$$s_\lambda(x_1, \cdots, x_n) s_\mu(x_1, \cdots, x_n) = \sum_{\nu \in \Upsilon} c^{\nu}_{\lambda\mu} s_\nu(x_1, \cdots, x_n) \tag{70}$$

$$s_{\nu/\lambda}(x_1, \cdots, x_n) = \sum_{\mu \in \Upsilon} c^{\nu}_{\lambda\mu} s_\mu(x_1, \cdots, x_n) \tag{71}$$

a formal proof of this can be straightforwardly given by applying the canonical projections π_n from Λ onto Λ_n

Following this trend, we prove the following identity for schur polynomials in Λ_{n+m}

Proposition 4.5.5. *The following identity for the schur polynomials in Λ_{n+m} hold:*

$$s_\nu(x_1,\cdots,x_n,y_1,\cdots,y_m)=\sum_{\lambda,\mu}c^{\nu}_{\lambda\mu}s_\lambda(x_1,\cdots,x_n)s_\mu(y_1,\cdots,y_m) \tag{72}$$

Proof. Note that, on the LHS we have the sum of all monomials of the form x^V, where V is a tableau with entries in $[n+m]$ of shape ν. Fix some V satisfying the latter specifications. Then, choose any $\lambda\subset\nu$. Then, divide up this tableau into two parts V/λ of shape ν/λ and V_λ of shape λ. Note that any integer in *content*(V/λ) is strictly greater than that of *content*(V_λ), since $\lambda\subset\nu$. This allows us to expand the RHS as follows

$$s_\nu(x_1,\cdots,x_n,y_1,\cdots,y_m)=\sum_{\lambda\subset\nu}s_\lambda(x_1,\cdots,x_n)s_{\nu/\lambda}(y_1,\cdots,y_m) \tag{73}$$

Applying 71 we are done. ■

We have now gained thoughtful understanding as how Littlewood-Richardson coefficients arise in relation to tableaux. Our next step will be to tie these concepts with the Hall inner product.

Recall that, under the hall inner product, the schur functions form an orthonormal basis for the ring of symmetric functions. Thus, any function $f\in\Lambda$, is uniquely determined by its hall inner product with some schur function s_λ. That is the linear coefficients $\{a_{\lambda\nu}\}_{\nu\in\Upsilon}$, that show up in $\langle f,s_\lambda\rangle=\sum_\nu a_{\lambda\nu}s_\nu$, completely determine f.

Now, as an exercise of thought, imagine that we weren't able to decide on what what the schur polynomials for **skew tableaux** should look like; that is, we could only define the schur polynomials for tableaux as in 30 but not its generalization 31. Then, the discussion in the preceding paragraph and 4.5.3 together would allow us to arrive at the definition gave in 31. Why? Let ν,λ,μ denote tableaux shapes. Consider the following chain of equalities, where we use 4.5.3 and 4.5.4:

$$\langle s_\nu,s_\lambda s_\mu\rangle=\langle s_\nu,\sum_{\alpha\in\Upsilon}c^{\alpha}_{\lambda\mu}s_\alpha\rangle=\sum_{\alpha\in\Upsilon}c^{\alpha}_{\lambda\mu}\delta_{\nu\alpha}=c^{\nu}_{\lambda\mu}=\sum_{\alpha\in\Upsilon}c^{\nu}_{\lambda\alpha}\delta_{\alpha\mu}=\langle\sum_{\alpha\in\Upsilon}c^{\nu}_{\lambda\alpha}s_\alpha,s_\mu\rangle=\langle s_{\nu/\lambda},s_\mu\rangle \tag{74}$$

From this equality, we see yet another consistent way of defining the skew schur polynomials. One could just declare $s_{\nu/\lambda}$ to be the **unique** symmetric function such that the equality above holds.

We have only scratched the surface of the diverse range of combinatorial applications related to these coefficients. For instance, without much additional work, we can prove the following about tableaux:

Proposition 4.5.6. *The following sets have the same cardinality $c^{\nu}_{\lambda\mu}$:*

1. $\mathcal{S}(\nu/\mu,T_0)$ *for any tableau T_0 on λ*
2. $\mathcal{T}(\mu,\lambda,V_0)$ *for any tableau V_0 on ν*
3. $\mathcal{S}(\tilde{\nu}/\tilde{\lambda},\tilde{U}_0$ *for any tableau $\tilde{U}_0$) on the conjugate diagram $\tilde{\mu}$*
4. $\mathcal{T}(\tilde{\lambda},\tilde{\mu},\tilde{V}_0$ *for any tableau $\tilde{U}_0$) on the conjugate diagram $\tilde{\nu}$*
5. $\mathcal{S}(\lambda*\mu,V_0)$ *for any tableau V_0 on ν.*

Proof. Note that 1 and 2 have the same cardinality by 4.5.2, simply by switching the roles of λ and μ. For (3), simply note that every element of $\mathcal{S}(\nu/\mu, U_0)$ can be made into a unique element of $\mathcal{S}(\tilde{\nu}/\tilde{\lambda}, \tilde{U}_0)$ by taking its transpose and vice versa. Thus, indeed (3) has cardinality $c^{\nu}_{\lambda\mu}$; the latter then implies (4) has the same cardinality by applying 4.5.2 again.

For (5), recall that:

$$\mathcal{T}(\lambda, \mu, V_0) = \{\text{pairs of tableaux (T,U)} \,|\, T\, on\, \lambda, U\, on\, \mu,\ and\ T \cdot U = V_0\}$$

Recall also that $T \cdot U$ is attained precisely by $Rect(\lambda * \mu)$ where $\lambda * \mu$ is filled with the entries of T and U respectively (as discussed in 2.7). But since,

$$\mathcal{S}(\lambda * \mu, U_0) = \{\text{skew tableaux } S \text{ on the shape } \lambda * \mu \,|\, Rect(S) = V_0\}$$

We have a one to one correspondence between these, from which we conclude (5) has the desired cardinality. ■

The latter result served to exemplify the endless possibilities for these coefficients. The theory surrounding these is very rich. For instance, they are of key importance for the results known as Littlewood-Richardson Rule and Frobenius reciprocity. Furthermore, they also aid in the understanding of the tensor product decomposition of Schur Modules, Newell-Littlewood numbers, and reduced Kronecker coefficients of the symmetric group. For a first exposition to these results, I would refer the reader to [1], chapters 7 and 8. For a more organic and modern approach, the reader should visit [9], Chapter 6.

References

[1] William Fulton, *Young Tableaux. With applications to representation theory and geometry*, London Math. Society Student Texts **35**, Cambridge University Press, 1997.

[2] Richard P. Stanley S. Fomin, *Enumerative Combinatorics, Volume 2*, Cambridge Studies in Advanced Mathematics, Cambridge University Press, 1999.

[3] Dan Romik, *The Surprising Mathematics of Longest Increasing Subsequences*, Institute of Mathematical Statistics Textbooks, Cambridge University Press, 2015

[4] Chen, et. al *Crossings and Nestings of Matchings and Partitions*, Trans. Amer. Math. Soc. **359**, ArXiv, 2007.

[5] C. Krattenthaler, *Growth diagrams, and increasing and decreasing chains in fillings of Ferrers shapes*, Adv. Appl. Math., **37**, 2006

[6] D. Armstrong, c. Stump, H. Thomas, *A uniform bijection between nonnestings and noncrossings partitions* Trans. Amer. Math. Soc., **365(8), 4121–4151, ArXiv, 2013**

[7] Adriano Garsia and A. Goupil *Character polynomials, their q-analogs and the Kronecker product* **Electronic Journal of Combinatorics, 16, 2009.**

[8] Bruce Sagan *The Symmetric Group: Representations, Combinatorial Algorithms, and Symmetric Functions* **Springer, New York, 2001**

[9] T. Silberstein, F. Sacarabotti, F. Tolli *Representation theory of the symmetric groups : the Okounkov-Vershik approach, character formulas, and partition algebras,* **Cambridge Studies in Advanced Mathematics, Cambridge University Press, 2010**

[10] Wikepedia. Inverse Limit — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Inverse_limit` [**Last Accessed, Monday, March 30th, 2022**]