PARTITIONS, TABLEAUX, PERMUTATIONS, OH MY!

DAN SHAW

1. Introduction

This paper considers the nature of partitions and many of the results surrounding them. Partitions lead into a discussion of Tableaux, the RSK algorithm, Euler's pentagonal number theorem and even symmetric polynomials. At first these topics seem to have little to do with each other but partitions tie them all together. I prove small results about each of those topics and even come up with a conjecture that I have yet to prove.

2. Partitions

The definition of tableau begins with the partition diagram for a particular partition λ. In order to define such a diagram, and in turn tableaux, the concept of a partition must be clear.

Definition 2.1. *Let n be a positive integer. A partition of n is an expression for n as a sum of positive integers, where the order of the summands is unimportant* [1].

For example, if $n = 5$ one partition of n is $4 + 1$. Note that the number of partitions of any given n is finite. In the case when $n = 5$ there exist precisely 7 partitions of n:

$$\begin{aligned} 5 &= 5 \\ &= 4+1 \\ &= 3+1+1 \\ &= 3+2 \\ &= 2+2+1 \\ &= 2+1+1+1 \\ &= 1+1+1+1+1. \end{aligned}$$

Notice that n is always a partition of itself. One of the most important initial problems surrounding partitions was to find a formla to count the number of different partitions that exist for each integer. The calculations necessary to obtain the number of partitions for small integers do not require much thought. As n gets larger the length of time necessary to find the number of partitions of n increases severely, even though the actual calculations are still relatively trivial. In the next section the construction of a generating function for partition numbers solves the problem of the amount of time required to compute answers. The coefficient on the n^{th} term in the generating function equals the number of partitions on n.

For the remainder of the paper $\lambda \vdash n$ signifies "λ is a partition of n." As simple as it is to write partitions out numerically, it is often useful to represent them graphically. Definition 2.2 explains the construction of the partition diagram for a given partition λ.

Definition 2.2. *Let λ be the partition $n = n_1 + \cdots + n_k$, with $n_1 \geq \cdots \geq n_k$. The diagram of λ has k rows; the i^{th} row (numbering from the top) contains n_i cells, aligned at the left. $D(\lambda)$ refers to the diagram of λ* [1].

The cells are represented by either dots or empty squares. Figure 1 gives an example of each type of digram for the partition of 8 where $\lambda = 4+2+1+1$. In general it will be more useful to represent partitions with the second type of diagram. This type of diagram leads to the definition of tableaux.

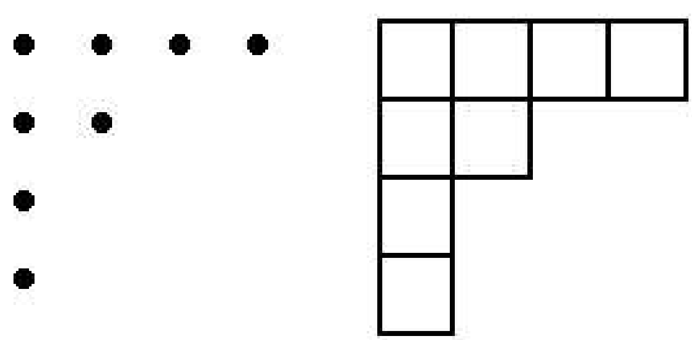

FIGURE 1. Partition Diagrams for $8 = 4 + 2 + 1 + 1$

A few proofs about the nature of partitions use the concept of a dual partition. Given a partition λ, we denote the dual partition or conjugate of λ by λ^*.

Definition 2.3. *Let $\lambda \vdash n$. The conjugate of λ is the partition of n whose diagram is the transpose (in the sense of matrices) of that of λ* [1].

Definition 2.3 can be stated another way. To construct the conjugate of a partition diagram we simply interchange the rows and columns of $D(\lambda)$. Clearly the dual partition λ^* is also a partition of n. There is still a total of n dots or cells in the diagram. If we take as λ the partition from Figure 1, notice that $\lambda = \lambda^*$. A more interesting example occurs when examining the partition of 9: $\lambda = 5 + 3 + 1$. Figure 2 shows $D(\lambda)$ and $D(\lambda^*)$. Then, λ^* represents the partition 3+2+2+1+1 which is also a partition of 9.

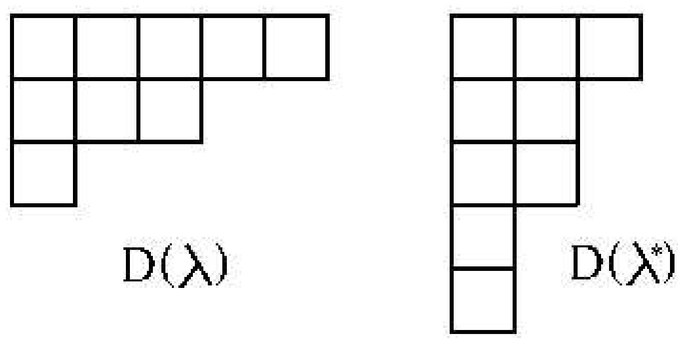

FIGURE 2. The Dual Partition of λ=5+3+1

3. Theorems

Theorem 3.1 appears as an exercise on page 33 of Yaglom's text [5].

Theorem 3.1. *The number of partitions of n into at most m parts is eqaul to the number of partitions of n with parts smaller than or equal to m* [5].

Suppose that m equals 3. Now consider the partitions of $n = 6$: 6, 5+1, 4+2, 4+1+1, 3+3, 3+2+1, 3+1+1+1, 2+2+2, 2+2+1+1, 2+1+1+1+1, and 1+1+1+1+1+1 are all of the partitions of 6. In order to make the statement of the theorem clearer we can sort these partitions into the appropriate subsets.

With at most 3 parts	$\{6,\ 5+1,\ 4+2,\ 4+1+1,$ $3+3,\ 2+2+2,\ 3+2+1\}$
With parts ≤ 3	$\{3+3,\ 3+2+1,\ 3+1+1+1,$ $2+2+2,\ 2+2+1+1,\ 2+1+1+1+1,$ $1+1+1+1+1+1\}$

These lists point out an important part of the theorem: the two sets are not necessarily disjoint. In other words the intersection of the two sets is potentially non-empty. Thus, the proof really only states the equivalence of the size of the two sets but says nothing about the total number of partitions of n. In order to prove theorem 3.1 we want to look at two subsets of the partitions of n and prove that they have to same size. In this case the best way to do so is to show that there exists a bijection between the two sets.

Proof. Start with $\lambda \vdash n$ where $\lambda = n_1 + n_2 + \cdots + n_k$ where $k \leq m$. λ is a generic partition of n with no more than m parts. In fact, λ could only have one part. Consider the partition diagram, D(λ), of λ. Each row of a partition diagram represents one element of a partition. Notice that $D(\lambda)$ has at most m rows. The conjugate of λ, or λ^* is also a partiton of n since there must also be n cells in $D(\lambda^*)$. Recall the definition of conjugate. We simply interchange the rows and columns of $D(\lambda)$. Thus, λ^* has at most m columns. In other words any particular row in $D(\lambda^*)$ has fewer than m cells. Once again, each row represents one part of the partition. So, each part of λ^* is less than or m.

Now that we have a way to get from a partition of n with at most m parts to a partition of n with parts less than or equal to m we need to prove that the sets of each are the same size. Suppose there exists a mapping, ϕ, that takes a partition diagram to its conjugate. Every partition diagram has a conjugate. In other words $\phi(D(\lambda))$ is onto. Furthermore, this mapping must be one-to-one. Given two distinct partitions of n, $\lambda_1 = a_1 + a_2 + \cdots + a_k$ and $\lambda_2 = b_1 + b_2 + \cdots + b_j$ can λ_1^* equal λ_2^*? Since λ_1 and λ_2 are distinct a_i cannot equal b_i for all i. In terms of diagrams, there must be at least two rows that differ between the two partitions. If λ_1^* equalled λ_2^* each column would be the same. This would imply that each row in the original diagrams were equivalent, further implying that $\lambda_1 = \lambda_2$. Thus the map must be one-to-one. A map that is both onto and one-to-one is a bijection. For each diagram with fewer than m rows, there is a conjugate with fewer than m columns. This bijection proves the equivalence of the size of the sets in question. □

The next result is again based on the size of certain subsets of partitions of n. This theorem also comes from page 33 of Yaglom's text [5]. Theorem 3.2 refers to partitions with distinct parts. A partition with distinct parts is one in which no two parts are equivalent. Be careful to differentiate this from *distinct partitions*. Distinct partitions are two partitions that are not entirely equivalent.

Theorem 3.2. *Suppose that $n > \frac{m(m+1)}{2}$. The number of partitions of n into m distinct parts is equal to the number of partitions of $n - \frac{m(m+1)}{2}$ into at most m (not necessarily distinct) parts* [5].

As with Theorem 3.1, the goal of the proof of Theorem 3.2 is to show the equivalence of the sizes of two subsets of the partitions of n. Once again, the existence of a bijective mapping between the subsets suffices to prove the theorem.

Proof. Suppose that $n > \frac{m(m+1)}{2}$. Also suppose that $\lambda \vdash n$ such that $\lambda = x_1 + x_2 + \cdots + x_m$ where $x_i > x_j$ when $i < j$. In other words x_1 is the largest part of λ. Since we know that $\sum_{i=1}^{m} i = \frac{m(m+1)}{2}$ we also know that

$$n - \frac{m(m+1)}{2} = (x_1 - m) + (x_2 - (m-1)) + \cdots + (x_m - 1).$$

We know that $x_m \geq 1$, thus $x_{m-1} \geq 2$, $x_{m-2} \geq 3$, $\ldots$, $x_1 \geq m$. Furthermore, $x_1 - m \geq x_2 - (m-1) \geq \cdots \geq x_m - 1 \geq 0$. In order to keep the notation simpler we need to define z_i:

$$\begin{aligned} z_1 &= x_1 - m \\ z_2 &= x_2 - (m-1) \\ \vdots z_m &= x_m - 1. \end{aligned}$$

Since From any partition of n, even if it has fewer than m parts, a partition of $n - \frac{m(m+1)}{2}$ can easily be constructed. Namely, $\lambda_n = z_1 + \cdots + z_i$ where z_i is the last nonzero z. This construction works for every partition of n with at most m distinct parts. Furthermore, λ_n must have at most m parts. Let A equal the set of partitions on n that have fewer than m distinct parts and let B equal the set of partitions on $n - \frac{m(m+1)}{2}$ into at most m parts. In order to prove the equivalence of $|A|$ and $|B|$ we need to be able to construct λ, a partition of n, from λ_n, a partition of $n - \frac{m(m+1)}{2}$. In other words we need a map, ϕ from A to B given by $\lambda_n \mapsto \lambda$. Given that $\lambda_n = z_1 + z_2 + \cdots + z_i$ we can simply add back the integers 1 through m. Thus, $n = (z_1 + m) + (z_2 + (m-1)) + \cdots + (z_i + (m-i-1)) + (0 + (m-i-2))) + \cdots + 1$. We should express those terms using the x_js from the original partition of n:

$$\begin{aligned}
x_1 &= z_1 + m \\
x_2 &= z_2 + (m-1) \\
&\vdots \\
x_i &= z_i + (m-i-1) \\
x_{i+1} &= 0 + (m-i-2) \\
&\vdots \\
x_m &= 0 + (m-(m-1)) = 1.
\end{aligned}$$

At first glance it may appear as though x_m will always equal 1. However, if $i = m$, then z_m is nonzero and thus $x_m = z_m + 1$. We can now construct a partition of $n - \frac{m(m+1)}{2}$ from a partition of n and *vice versa*. Recall that we started with a partition of n that had precisely m distinct parts. We ended up with a partition of $n - \frac{m(m+1)}{2}$ with at most m parts. The fact that ϕ is one-to-one and onto proves that A and B have the same size. Thus each set has the same number of elements. □

An argument for the same theorem can be made using partition diagrams alone. This proof shows two important facts about partitions and combinatorics in general. First of all, partition diagrams, as mentioned earlier, represent more than a convenient way to represent partitions. They often make for an elegant proof technique. More importantly, almost every proof in combinatorics has multiple methods. This means that problems that may have been solved a hundred years ago still prove to be intriguing. The goal now is to find the best method of proof. To that end, Theorem 3.2 is proven using partition diagrams below.

Proof. Begin with a partition of $n = \lambda = x_1 + x_2 + \cdots + x_m$ where $x_i > x_j$ when $i < j$. In other words x_1 is the largest part of λ. $D(\lambda)$ necessarily has an equilateral triangle of cells embedded in the left side, with one of the sides being on the left edge. We know that such a triangle will exist since the parts of λ are distinct. Thus, each one is larger than the one below it. By removing this triangle we subtract each number from 1 to m and are therefore, by the work above, left with a partition of $n - \frac{m(m+1)}{2}$. Each diagram of a partition of n must have such a triangle and is therefore paired with a particular partition of $n - \frac{m(m+1)}{2}$. Moreover, if a triangle is added to a partition of $n - \frac{m(m+1)}{2}$

we obtain a partition of n. Thus the set of partitions of n into m distnct parts is equal to the number of partitions of $\frac{m(m+1)}{2}$. □

Since this proof relies heavily on the graphical analysis of partition diagrams an example helps to clarify the method. Figure 3 contains two partitions. The first is a partition of 15: λ=6+4+3+2. The second partition results from removing the triangle highlighted in the first. This new partition is a partition of 5. In this example n=15 and m=4. Note that $D(\lambda_1)$ is a partition of n and $D(\lambda_2)$ is a partition of $n - \frac{m(m+1)}{2}$.

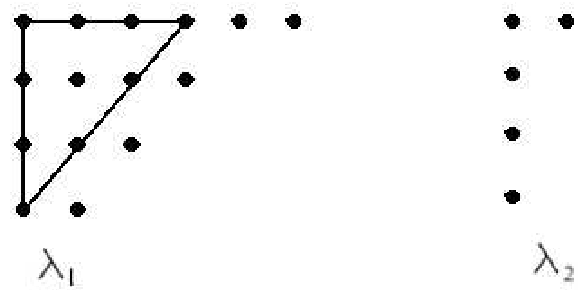

Figure 3. The graphical proof of Theorem 3.2

Already we have a few different methods to use to best prove results about partitions. The next theorem gives us yet another tool.

4. The Generating Function for Partition Numbers

Generating functions have a very broad, simple definition. This simplicity allows for a usefulness in nearly all areas of mathematics. The field of Combinatorics is no exception. In order for the generating function to be useful we must define partition number.

Definition 4.1. *A partition number, $P(n)$, is the total number of partitions of n.*

As mentioned above, one of the biggest questions surrounding partitions is exactly how many different partitions there are for a given n. The generating function for partition numbers answers this question.

Definition 4.2. *A generating function is a power series whose coefficients describe the nature of a specific set.*

On page 38 of Cameron's book he defines the generating function for partition numbers as follows:

$$\sum_{n\geq 0} P(n)t^n = \prod_{i=1}^{n} \frac{1}{1-t^i} \tag{1}$$

with the convention that $P(0) = 1$ [1]. Theorem 4.1 more clearly states the significance of the generating function.

Theorem 4.1. *The number of partitions of n is equal to the coefficient on t^n in the product*

$$\prod_{i=1}^{n} \frac{1}{1-t^i}.$$

We want to prove that the coefficient on t^n in the series in Theorem 4.1 equals the number of partitions of n. After some expansion of the product in Theorem 4.1, the equivalence of Equation 1 is proven combinatorially.

Proof. Notice that the right side of Equation 1 is in the form of a product of gometric series. That fact implies that the equation can be expanded in the following way:

$$\prod_{i=1}^{n} \frac{1}{1-t^i} = \prod_{i=1}^{n} (1 + t^i + t^{2i} + t^{3i} + t^{4i} \dots) \tag{2}$$

$$= (1 + t + t^2 + \dots)(1 + t^2 + t^4 + \dots) \cdots . \tag{3}$$

In order to see the source of the partition numbers, we need to first look at t^n. A term that has the n^{th} power in the expansion of Equation 3 is obtained by selecting t^{1a_1} from the first factor, t^{2a_2} from the second, and so on, where $a_1 + 2a_2 + 3a_3 + \cdots = n$. In other words, a particular t^n is constructed by multiplying different powers of t that sum up to n. By definition we know that

$$\underbrace{1 + 1 + \cdots + 1}_{a_1} + \underbrace{2 + 2 + \cdots + 2}_{a_2} + \underbrace{3 + 3 + \cdots + 3}_{a_3} \cdots \vdash n$$

.

Notice that every partition of n is created by selecting values of t from different factors of Equation 3. All of the ones are represented in the first factor, twos in the second, threes in the third, and so on. Every t^n that comes out of the expansion of Equation 3 must come from a partition on n. The exponent results directly from the summation of integers smaller than or equal to n. Also notice that every partition of n appears in the product. This ensures that all partitions of n are generated. Thus the coefficient on t^n does in fact equal $P(n)$ [1]. □

Using $n = 3$ as an example, some of the details of the proof become clearer. There are only 3 different partitions to be generated: $\lambda_1 = 3$, $\lambda_2 = 2 + 1$, and $\lambda_3 = 1 + 1 + 1$. With $n = 3$ the pertinent coefficient is that on t^3. In other words the number of t^3s left after expansion should equal the number of partitions of 3. Moreover, the way the product creates cubes of t should reflect the three partitions of 3.

How many ways are there to multiply t^i to obtain t^3 where i is a positive integer? The first is $t^3 \cdot t^0$. In the product above $t^3 \cdot t^0$ appears twice: once when the t^3 in the first factor of the product in Equation 3 term is multiplied by ones in every other factor and again when the t^3 in the third factor is multiplied by ones in every other factor. These two products represent λ_3 and λ_1 respectively. The other way to multiply powers of t to obtain t^3 is $t^2 \cdot t^1$. This product only comes up once in the series: when the t in the first factor is multiplied by the t^2 in the second factor and by ones everywhere else. This product represents the partition λ_2.

The power of the generating function for partition numbers lies in its usefulness in proofs. The proof of theorem 4.2 highlights that ability.

Theorem 4.2. *Given the set of all partitions of n and some positive integer k the subset of partitions whose parts occur no more than $k-1$ times is equal to the subset of partitions whose parts k does not divide* [5].

For example, if n equals 5 and k equals 2 the above constraints would leave the following sets:

$$\{5,\ 4+1,\ 3+2\}$$

and

$$\{5,\ 3+1+1,\ 1+1+1+1+1\}$$

Each set has exactly 3 elements. Is this equality true for all k and all n? In other words, is the set of partitions of n whose parts occur no more than $k-1$ times the same size as the set of partitions of n whose parts k does not divide? The proof of theorem 4.2 expresses each set using the generating function and then shows that the generating functions for each set are equivalent.

Proof. We should begin by modifying the generating function for all partition numbers in order to just count the partitions we want. Let's start with partitions of distinct parts. A partition with distinct parts is one in which no integer part is repeated. Recall that each integer i in a given partition is counted by the i^{th} factor in equation 3. We only want to consider one element of each factor. Thus, the generating function for partitions with distinct parts is

$$(1+t)(1+t^2)(1+t^3)\cdots.$$

This is a special case of the problem at hand, namely where $k = 2$. Generalizing for $k-1$, the expression for partitions with parts appearing no more than $k-1$ times is

$$(1+t+\cdots+t^{k-1})(1+t^2+\cdots+t^{2(k-1)})(1+t^3+\cdots+t^{3(k+1)})\cdots.$$

Suppose $\lambda \vdash n{=}a_1+a_2+\cdots+a_m$ where no a_i appears more than $k-1$ times. The repetition of each term a_1 in the partition is determined by the power on t in the a_1^{th} factor of the term above. In each factor of that function there is at most $k-1$ terms. Thus no part can be chosen more than $k-1$ times. The expression for parts not divisible by k is slightly more complicated, but follows the same sort of idea as above. We want to only keep the parts that k does not divide. We can force this to happen by taking out all multiples of k. The generating function for parts not divisible by k is a product of the following:

$$\begin{aligned}
(1+t+t^2+\dots)\\
(1+t^2+t^4+\dots)\\
\vdots\\
(1+t^{k-1}+t^{2(k-1)}+\dots)\\
(1+t^{k+1}+t^{2(k+1)}+\dots)\\
\vdots\\
(1+t^{2(k-1)}+t^{2(2k-1)}+\dots)\\
(1+t^{2(k+1)}+t^{2(2k+1)}+\dots)\\
\vdots
\end{aligned}$$

This function represents partitions with parts not divisible by k because a part that is divisible by k must come from a coeffecient on t^i where $k|i$. All of the factors in the product above are not divisible by k. Essentially, equation given by multiplying the above factors is the generating function for partition numbers with factors that begin with t^{mk} removed.

Now, to show that the two expressions are actually equivalent, we want to put them back into a more manageable form. To do this we can multiply the second function by 1 in a sufficienty creative manner. Take the generating function for parts not divisible by k and multiply by

$$\frac{1-t}{1-t}\cdot\frac{1-t^2}{1-t^2}\cdots\frac{1-t^{k-1}}{1-t^{k-1}}\cdot\frac{1-t^{k+1}}{1-t^{k+1}}\cdots. \tag{4}$$

Note that Equation 4 can be rewritten in a more suggestive manner:

$$(1-t)\cdot\frac{1}{1-t}\cdot(1-t^2)\cdot\frac{1}{1-t^2}\cdots(1-t^{k-1})\cdot\frac{1}{1-t^{k-1}}\cdots \tag{5}$$

Now we can see that Equation 5 contains gemoetric series, moreover, those series are exactly those series in the generating function for parts not divisible by k. Thus, upon multiplying that generating function by Equation 5 we are just left with

$$(1-t)(1-t^2)\cdots(1-t^{k-1})(1-t^{k+1})\cdots(1-t^{2(k-1)})\cdots \tag{6}$$

Recognize that upon expansion Equation 6 simplifies to the generating function for partitions with parts appearing no more than $k-1$ times. The equivalence of the two generating functions proves the equivalence of the size of the two sets. □

The next result is a variation on a proof above. Specifically, this new proof highlights the power of partition diagrams. Theorem 4.2 proved that the number of partitions of n into parts that occur no more than $k-1$ times is equal to the number of partitions of n whose parts k does not divide. Notice a particular case of this theorem: when k=2. The partitions of n are now split up into one set where each partition has entirely distinct parts and the other set where each partition is composed of odd parts. The proof of Theorem 4.2 proves this claim, but a proof based on partition diagrams yields the same result. Figure 4 is an image of both the distinct and odd partitons of 10. It is apparent that each set consists of ten elements. But why is this the case for every n? The arrows in the diagram represent an algorithm that maps an odd partition to a partition of distinct parts. The algorithm can be stated as follows, where $D(\lambda)'$ is the transformation of $D(\lambda)$:

Step 1: Examine $D(\lambda)$: If $D(\lambda)$ has distinct parts, $D(\lambda)' = D(\lambda)$. If not, procede to *Step* 2.

Step 2: Take repeated parts not equal to one and combine them by addition. If the partition still does not have distinct parts procede to

Step 3.

Step 3: Take repeated ones and add them into the highest power of two possible. Repeat *Step* 3 until $D(\lambda)'$ is composed of distinct parts.

The algorithm works similarly in the other direction, from partitions of distinct parts to odd partitions:

Step 1: Examine $D(\lambda)$, if $D(\lambda)$ is composed of odd parts, $D(\lambda)' = D(\lambda)$. If not, procede to *Step* 2, but leave all odd parts intact.

Step 2: Find all parts, a_i of $D(\lambda)$ such that $a_i = 2^m$ for some integer m. Break all such a_i into parts of size one. If $D(\lambda)$ is still not composed of odd parts, proced to *Step* 3.

Step 3: Any left over part, a_j is necesarrily of the form $k2^m$ for some integers k and m. Now take a_j and break into two halves. If $D(\lambda)'$ is still not composed of odd parts, repeat *Step* 3.

The goal of such an algorithm is to show a bijection between the two sets. In other words, to show that a member of one set corresponds to exactly one member of the other set. This algorithm does exactly that. The process is simply reversed to go from one set to the other. Thus, each set must have the same number of elements. Figure 4 shows the algorithm in action. In this example λ=5+3+3+3+1+1+1+1+1+1 and thus is an odd partition. We want to use the algorithm to obtain a partition composed of distinct parts. We may procede to step 2 since λ is not made up of distinct parts. Step 2 tells us to combine repeated parts not equal to 1. The only such part is equal to 3. Thus we combine two of the threes and are left with a part of equal to 6 and one of the original threes. Now the only repeated parts are of size one, thus we can move on to step 3. There are 6 ones in λ. Step 3 says to add them into the highest power of 2 possible. In this case that power is 2. Thus, we combine four of the ones and now have a part of size 4 and two ones left over. Since we still have repeated ones, we repeat step three and combine the last two ones and get a part of size two. After all of these steps we obtain the partition 20=6+5+4+3+2. Notice also that λ is a partition of 20 as well.

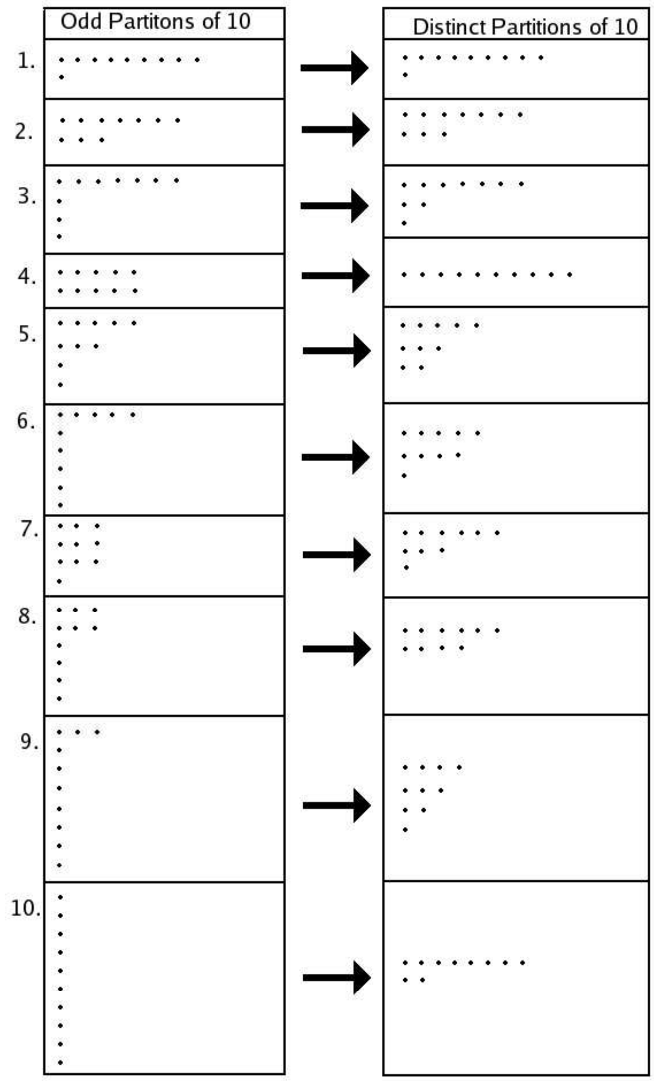

FIGURE 4. The odd and distinct partitions of 10

5. Tableaux

The theorems surrounding partitions show many interesting and even surprising properties. Examining partition diagrams shows that they are filled with unique properties as well. Recall that there were two

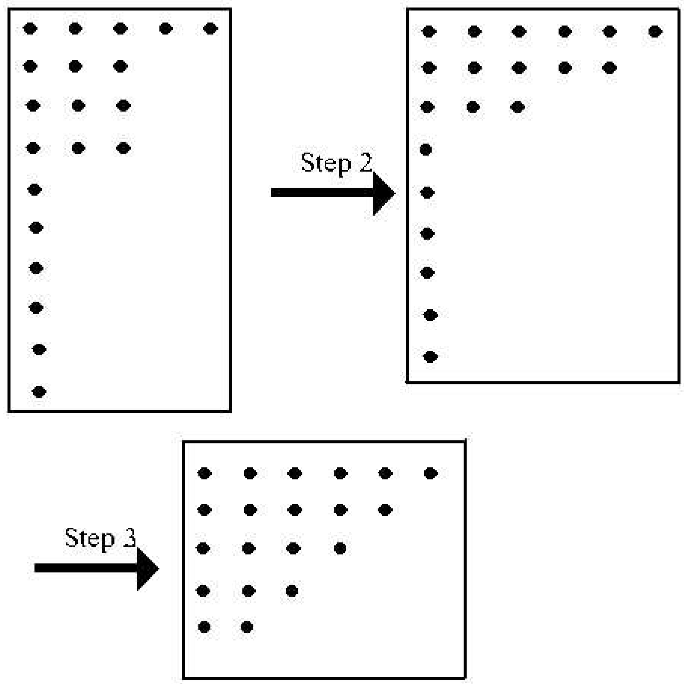

FIGURE 5. The Algorithm applied to λ=5+3+3+3+1+1+1+1+1+1

ways to represent partition graphically: with dots and with cells. Calling the spaces cells suggests that they could be filled with something. Indeed, when those cells are filled with the numbers 1 through n the partition diagram becomes a tableau.

Definition 5.1. *A tableau consists of a partition diagram with its cells numbered from* 1 *through* n *in any order* [2].

The partition λ=4+2+1+1 came up earlier in the example of partition diagram. Figure 6 gives an example of a tableau on λ.

Importantly there are no restrictions on the way the numbers can fill the cells. This makes counting the number of tableaux of a given shape relatively simple. There are n choices for the first cell, $n-1$ for the second, $n-2$ for the third, and so on. Thus, there are $n!$ different tableaux for each partition diagram on n. What if we put certain restrictions on the way that tableaux were filled?

Definition 5.2. *A standard tableau is a specific type of tableau in which the numbers 1 through n increase moving right and down the partition diagram.*

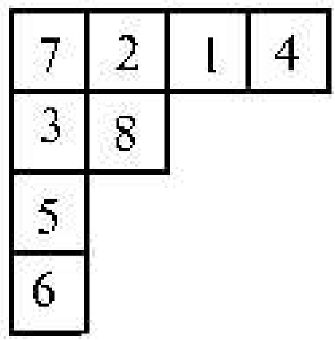

FIGURE 6. An example of a Tableau

Figure 7 gives an example of a standard tableau where n is equal to 13 and λ is the partition 5+3+2+2+1. Obviously this isn't the only example of a standard tableau for λ: the 13 and the 10 could be switched, or many other changes. Unfortunately, counting the number of standard tableaux of a given shape is not as simple as counting the number of tableaux.

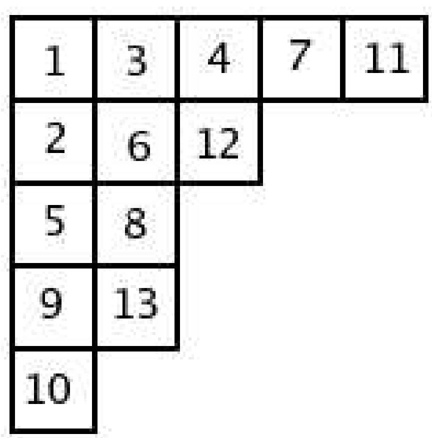

FIGURE 7. An example of a standard tableau with n=13

Certain parts of the calculation are easy to determine. For instance, 1 always must go in the top left hand corner of the standard tableaux. If 1 were in any other location the rule defining standard tableaux would necessarily be violated: the numbers would decrease moving either right or down somewhere. After this point, however, the calculations become very dependent on the individual partition diagram. As it turns out, there is no known easy combinatorial method to count the number of standard tableaux of a given shape. But, there is a very elegant and simple formula involving the hook-lengths of the cells of a partition diagram.

Definition 5.3. *The hook-length $h(i, j)$ of any cell (i, j) in a partition diagram is the number of cells directly left and below that cell, counting the cell itself* [1].

Return again to the partition of $n = 8 = 4 + 2 + 1 + 1$. Figure 8 displays the hook-lengths of each cell according to Definition 5.3. In particular the hook-length of cell (1, 1), or $h(1, 1)$, is equal to 7.

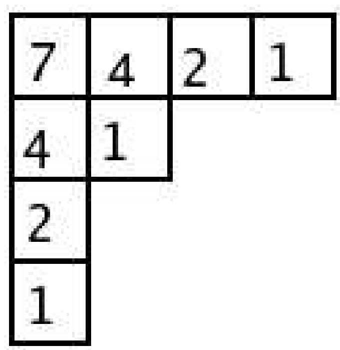

FIGURE 8. The Hook-Lengths for $\lambda = 4 + 2 + 1 + 1$

Theorem 5.1 states the hook-length formula. The formula counts the number of standard tableaux of a given shape using the product of the hook-lengths for that shape.

Theorem 5.1. *The number of standard tableaux of λ, denoted f_λ, where $\lambda \vdash n$ is equal to*

$$\frac{n!}{\prod_{(i,j)\in D(\lambda)} h(i,j)}.$$

Theorem 5.1 was first proven in 1953 [2]. Since no combinatorial method of proof is known, the proofs are all constructive in nature. In other words each proof createss an algorithm that builds standard tableaux out of tableaux. The formula for the number of tableaux is simple. In Feldman's text the proof centers on probability theory.

The backbone of Feldman's algorithm is that it generates a standard tableaux at random. By at random, Feldmen means that his algorithm will generate all standard tableaux with equal likelihood [2]. Feldman begins by defining a lower-right-hand-corner cell of a partition diagram.

Definition 5.4. *A lower-right-hand-corner-cell, or l_c, is a cell in a partition diagram that has no cells to the right or below it* [2].

In order for the diagram to be a standard tableaux n must be in a l_c. Otherwise, some number $k < n$ will fill some cell below or to the right of n. This violates the definition of a standard tableaux. Thus, the algorithm is as follows:

Step 1 : Begin by picking a cell C_1 in the partition diagram at random (i.e., all cells have equal probability). If we choose an l_c from the start, we immediately use that cell to hold the number n . If not, procede to *step* 2

Step 2 : Since C_1 is not an l_c, randomly pick a cell C_2 in the hook of C_1. If C_2 is an l_c then place n in C_2. If not, repeat *step* 2 until we finally pick some l_c as our C_j, for some $j < n$. Then n is put into cell C_j [2].

Step 3 : Repeat *steps* 1 and 2, this time considering $n - 1$ instead of n. Ignore cell C_j, as it is already filled.

The algorithm above always produces a standard tableau. After the cell for n is chosen, the algorithm resets and considers a new partition diagram for $n - 1$: the cell filled with n is effectively removed from the diagram. The numbers 1 through n will be increasing as we move right and down the diagram, since the higher numbers always fill the l_cs first. Furthermore, note that every standard tableaux of a given shape can be generated by this algorithm. n can fill any l_c, then $n - 1$ can fill any l_c of the new diagram, and so on. The crux of the proof comes in showing that all standard tableaux of a given shape appeard with equal probability. The statement of the algorithm gives an idea as to how theorem 5.1 is proven. Unfortunately, the scope of this paper limits a complete explination of the proof.

6. The RSK Algorithm

Standard tableaux certainly have a combinatorial feel about them, but they do not seem to have the normal connections to permutations and other counting methods. The RSK algorithm gives a way to generate standard tableaux from permutations. The algorithm leads to the RSK correspondence. The correspondence relates a permutation to two standard tableaux of the same shape.

In order to understand the RSK algorithm, a basic understanding of permutations is necessary. As the name suggests a permutation *permutes* the elements of a set. In other words a permutation on the set S will take each element to another member of the same set.

Definition 6.1. *A permutation is a rearrangement of the elements of an ordered set, S, into a one-to-one correspondence with S itself* [1].

Let's start with some basic examples. Consider the set S_1={2,1}. There are only 2!=2 permutations of S_1, namely {1,2} and {2,1}. Note that the set itself is considered a permutation. This permutation is usually called the identity permutation. Now consider the set S_2={1,2,3}. There are 3!=6 permutations of S_2: {1,2,3}, {1,3,2}, {2,1,3}, {2,3,1}, {3,1,2}, and {3,2,1}. Counting permutations is relatively easy and is based on the size of S. If $|S|=n$ then there are $n!$ permutations of S. There are n choices for the first spot, $n-1$ for the second, and so on. Multiplying these options together yields $n!$.

There are a few conventional ways to write down permutations, however, one is more helpful when using the RSK algorithm. This notation uses a $2 \times n$ matrix to represent a permutation. The first row is the original set S and the second row is the new arrangment of S. If S equalls $\{1, 2, 3, 4\}$ the matrix

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 3 & 1 \end{bmatrix}$$

represents the permutation {2,4,3,1}. Now that we have a basic understanding of permutations the RSK algorithm is as follows.

Given a permutation $g=(a_1, \ldots, a_n)$ on the set $N=\{1,2,\ldots,n\}$ we can build two standard tableaux (S, T) using this alorithm. First we have to create a subroutine called INSERT where the integer a is placed in the j^{th} row of a partial tableau T:

- If a is greater than the last element of the j^{th} row, then append it to this row. (If the j^{th} row is empty, put a in the first position.)
- Otherwise, let x be the smallest element of the j^{th} row for which a is not greater than x. 'Bump' x out of the j^{th} row, replacing it with a: then INSERT x into the $(j+1)^{st}$ row.

Using this subroutine we can give a complete definition for the RSK algorithm:

Start with S and T empty. For i=1,...,n, do the following:
(1) INSERT a_i into the first row of S. This causes a cascade of 'bumps,' ending with a new cell being created and a number (not exceeding a_i) written into it.
(2) Now creat a new cell in the same position in T and write i into it.

To help understand the algorithm Figure 9 shows an example of the RSK algorithm in action. Figure 10 gives examples of the RSK algorithm with some permutations on the set $\{1, 2, 3, 4, 5, 6, 7\}$ where a=(1,3,5,4,7,6,2) and b=(4,6,2,7,3,5,1).

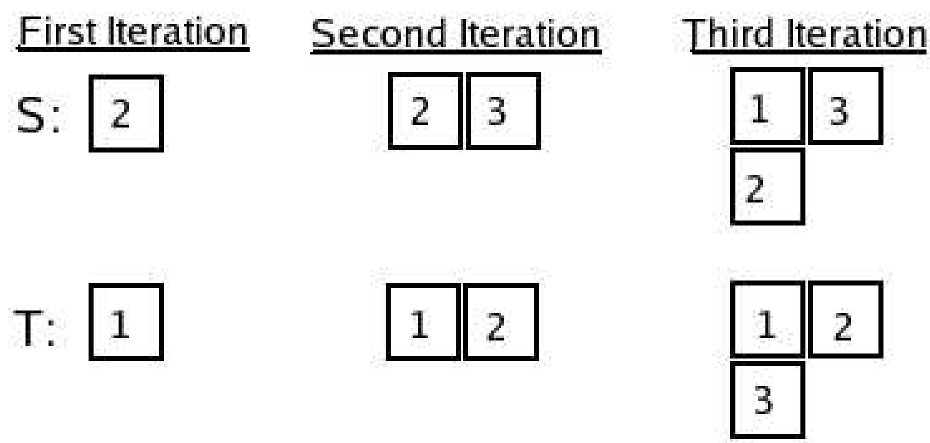

FIGURE 9. A quick application of the RSK algorithm where $g = (2, 3, 1)$ [1]

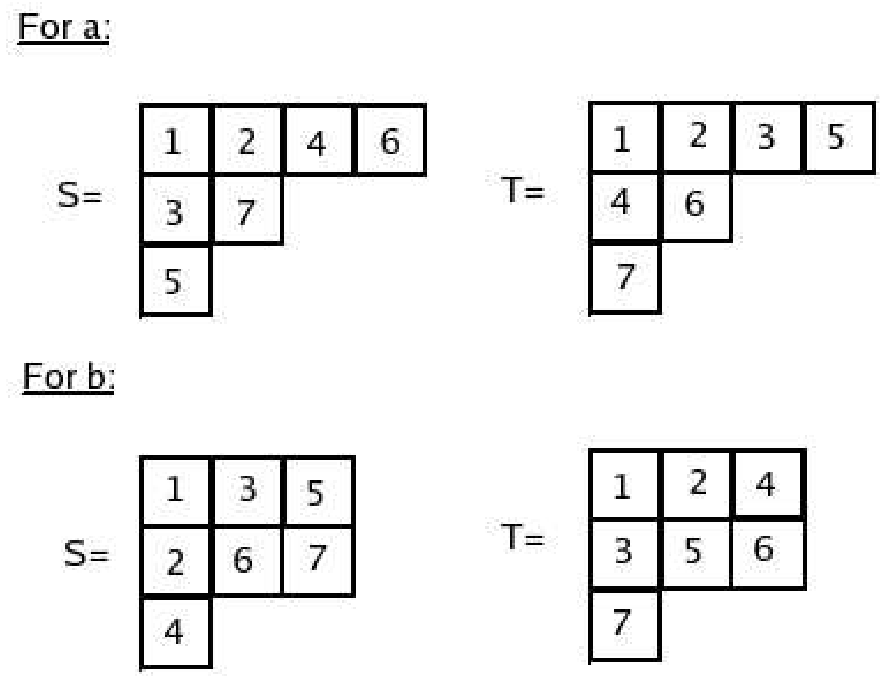

FIGURE 10. The pairs of Tableaux for $a = (1, 3, 5, 4, 7, 6, 2)$ and $b = (4, 6, 2, 7, 3, 5, 1)$

7. Theorems Surrounding the RSK Algorithm

The RSK algorithm gives a relationship between tableaux and permutations. By playing around with the RSK algorithm it became apparent that there potentially existed a relationship between the number of permutations of n and the number of standard tableaux of size n. The actual correspondence is as follows:

Conjecture 7.1. *The total number of permutations on the set* $\{1, 2, \ldots, n\}$ *can be expressed using the number of standard tableaux on* n. *In particular the sum of the squares of the number of standard tableaux for a given shape is equal to* $n!$.

In my own research I was both unable to prove Conjecture 7.1 myself and was unable to find any proof of the conjecture in any articles I read. That being said I am confident that the conjecture holds. Cameron uses the conjecture on page 138 of his book in order to prove a different result [1]. His passing reference does not give any hint as to how the conjecture could be proved, nor when it was proved. The following paragraphs illustrate Conjecture 7.1 for two small values of n.

Take a look at figures 11 and 12, where n=3 and 4. The figures each contain the different shapes of tableau possible for n=3 and 4. The hook length appears in each cell.

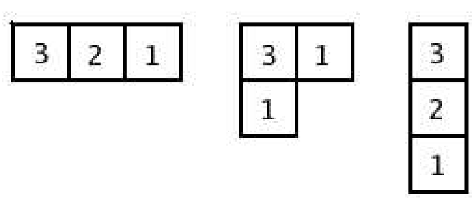

Figure 11. The three partition diagrams for n=3

Theorem 7.1 grounds tableaux, permutation, and the RSK algorithm in one of the most basic combinatorial problems: what is the sum of the first n integers? The cases for n=3 and 4 give an idea of what the theorem really states.

Recall that the hook length formula gives an expression for the total number, f_λ of standard tableaux of a given shape λ. The hook length formula is

$$f_\lambda = \frac{n!}{\prod_{(i,j)\in\lambda} h(i,j)}$$

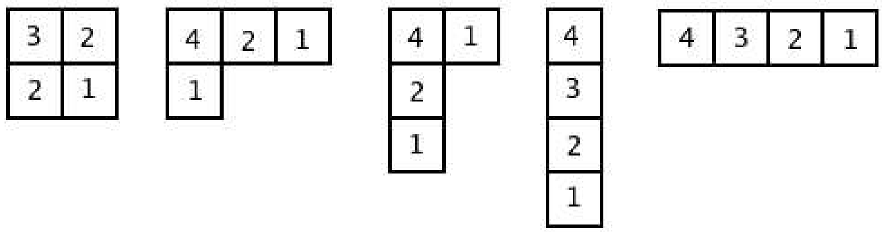

FIGURE 12. The five partition diagrams for n=5

where $h(i,j)$ represents the hook length of a given cell (i,j). Now we can apply the hook length formula to calculate the total number of standard tableaux of each shape. Figures 11 and 12 make the hook length calculations simple. For the sake of clear notation, $\lambda_{(n)}$ equals the total number of standard tableaux for an integer n of shape λ. Also, $P(n)$ equals the number of partitons of n. For n=3 the following calulations ensue:

$$\begin{aligned}
\lambda_{(3)} &= \sum_{a=1}^{P(3)} \frac{n!}{\prod_{(i,j)\in\lambda_a} h(i,j)} \\
&= \sum_{a=1}^{3} \frac{3!}{\prod_{(i,j)\in\lambda_a} h(i,j)} \\
&= \frac{6}{3\cdot 1\cdot 1} + \frac{6}{3\cdot 2\cdot 1} + \frac{6}{3\cdot 2\cdot 1} \\
&= 2+1+1
\end{aligned}$$

For now let's leave the sum in that form. Applying the same formula where n=4, we obtain the following:

$$\begin{aligned}
\lambda_{(4)} &= \sum_{a=1}^{5} \frac{4!}{\prod_{(i,j)\in\lambda_a} h(i,j)} \\
&= \frac{24}{3\cdot 2\cdot 2\cdot 1} + \frac{24}{4\cdot 2\cdot 2\cdot 1} + \frac{24}{4\cdot 2\cdot 2\cdot 1} + \frac{24}{4\cdot 3\cdot 2\cdot 1} + \frac{24}{4\cdot 3\cdot 2\cdot 1} \\
&= 2+3+3+1+1
\end{aligned}$$

According to Conjecture 7.1, we want to show that $\sum_1^p(\lambda_a)^2{=}n!$. Particularly for the cases when n=3 and n=4. For n=3,

$$\begin{aligned}\sum_1^P(\lambda_a)^2 &= 2^2+1^2+1^2\\ &= 4+1+1\\ &= 6\\ &= 3!\end{aligned}$$

For n=4

$$\begin{aligned}\sum_1^P(\lambda_a)^2 &= 2^2+3^2+3^2+1^2+1^2\\ &= 4+9+9+1+1\\ &= 24\\ &= 4!\end{aligned}$$

Based on the fact that the base cases work for n=1,2,3, and 4, induction seeems like an appropriate approach. Unfortunately, I have been unable to generate a proof of theorem 7.1.

The RSK algorithm appears to generate every tableau of size n. In other words, every λ_a can be generated by some permutation of n. Suppose we have a tableau λ on n such that its columns are a_1, a_2, $\ldots, a_r$ where r is the number of columns in λ. The theorem also utilizes the following notation: a_i^{-1} represents the elements in column a_i read from bottom to top. Formally, this idea is stated as follows:

Theorem 7.1. *The permutation* $\binom{1\ 2\ \ldots\ n}{a_1^{-1}\ a_2^{-1}\ldots a_r^{-1}}$ *will generate the pair of tableaux* (S,T) *where* S *has columns* a_1 *through* a_i*. Since every tableau can be written this way every tableau appears in the first position of the RSK algorithm.*

The proof of this theorem is straight forward.

Proof. Given a tableau λ with n entries, let a_i represent the i^{th} column of λ. Further, suppose that λ has r columns. Now examine the RSK algorithm as applied to the permutation $\binom{1\ 2\ldots n}{a_1^{-1}\ a_2^{-1}\ldots a_r^{-1}}$. Note that the first part of the algorithm will deal with a_1^{-1}. Since λ is a tableau the elements of a_1 are stricly increasing. Thus, the elements of a_1^{-1} are stricly decreasing. When a permutation decreases, a bump occurs in the RSK algorithm. Applying the RSK algorithm to this first set of numbers creates a bump at every step of the way and it will necessarily generate the first column of λ. Obviously this is true of each a_i^{-1}.

However, a problem occurs between a_i^{-1} and a_{i+1}^{-1}, and particularly between members of the same row. If an element in the j^{th} row of a_{i+1} is less than a member of the j^{th} row of a_i, the bump will occur in a_i rather than in a_{i+1} and the algorithm will generate a new tableau. But, since λ is a tableau, there is no such j. By definition each row must be increasing as we move right. Since a_{i+1} is to the right of a_i, each element in a_i must be smaller than the element in the same row of a_{i+1}. Thus, every tableaux can be generated by the RSK algorithm. □

8. Euler's Pentagonal Numbers Theorem

First of all, pentagonal numbers need to be defined. A *pentagonal number* is a number belonging to the following set:

$$\left\{\frac{k(3k-1)}{2} | k \in \mathbb{Z}\right\}.$$

Figure 13 shows the motivation for the name "pentagonal numbers." Each pentagonal number generated by a positive value of k can be drawn in a pentagonal shape like those in Figure 13. The numbers generated by when k is negative are more accuurately called generalized pentagonal numbers because they do not follow the pattern of the pentagonal numbers in Figure 13.

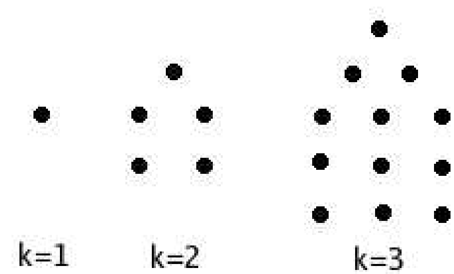

Figure 13. Small petagonal numbers [1]

Each of the three numbers, 1, 5 and , 12 take the shape of a pentagon when arranged as in Figure 13. Every pentagonal number generated by a positive k value can be drawn in in this fashion. The pentagon representing the k^{th} pentagonal number is constructed by adding one to each edge of the $(k-1)^{th}$ pentagon. What could pentagonal numbers possibly have to do with partitions and standard tableaux? Euler developed the following theorem:

Theorem 8.1. *(a) If n is not a pentagonal number, then the number of partitions of n into an even number of distinct parts and the number of partitions of n into an odd number of distinct parts are equal.*

(b) If $n = \frac{k(3k-1)}{2}$ for some $k \in \mathbb{Z}$, then the number of partitions of n into an even number of distinct parts exceeds the number of partitions into an odd number of distinct parts by one if k is even, and vice versa if k is odd [1].

Suppose that $n = 6$ [1]. There are four partitions of 6 into distinct parts, namely:

$$\begin{aligned} 6 &= 5+1 \\ &= 4+2 \\ &= 3+2+1. \end{aligned}$$

Notice that there are two partitions of each parity, i.e. two with an even number of elements and two with an odd number of elements. Based on the definition above, 7 is a pentagonal number with $k = -2$. If $n = 7$ there are five partitions of n into distinct parts:

$$\begin{aligned} 7 &= 7 \\ &= 6+1 \\ &= 5+2 \\ &= 4+3 \\ &= 4+2+1. \end{aligned}$$

Of these five, three have an even number of parts and two have an odd number.

The most obvious way to go about proving Euler's pentagonal number theorem is to attempt to produce a bijection between partitions with an even and an odd number of distinct parts, and examine what happens with the unique cases of pentagonal numbers. In Cameron's text he defines two new terms that help motivate such a bijection [1].

If we let λ be any partition of n into distinct parts we can define two subsets of the diagram $D(\lambda)$ as follows:
(1) the *base* is the bottom row of the diagram (i.e. the smallest part), and

(2) the *slope* is the set of cells starting at the right end of the top row and procedeing in a down and to the left for as long as possible [1]. Figure 14 displays these two definitions in a partition diagram.

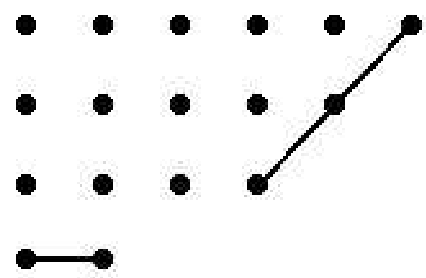

FIGURE 14. Base and slope [1]

Using these two definitions Cameron creates the following three classes of partitions of n with distinct parts:

Class 1 consists of the partitions for which *either* the base is longer than slope and they don't interesct, *or* the base exceeds the slope by at least 2;

Class 2 consists of the partitions for which *either* the slope is at least as long as the base and they don't intersect, *or* the slope is strictly longer than the base;

Class 3 consists of all other partitions with distinct parts.

Figure 15 gives examples of a partition from each of the three classes.

Armed with these classes and new definitions we can now prove Theorem 8.1; Euler's pentagonal numbers theorem. The proof hinges on the ability to create one partition diagram out of another. Thus, without further ado:

Proof. Given a partition λ in Class 1, we create a new partition λ' by removing the slope of λ and installing it as a new base. Assume that the slope of λ contains k cells. Thus we remove a cell from each of the k largest rows and create a new, smallest row of size k. By the definition of partitions in Class 1, λ' is a legal partition with distinct parts. Since the parts began distinct, removing the slope cannot make

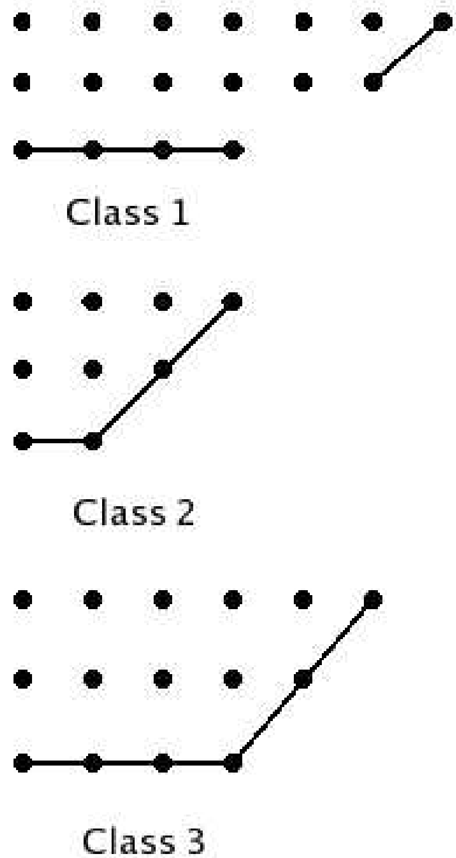

FIGURE 15. Top: Class 1 partiton, Middle: Class 2 partition, Bottom: Class 3 partition [1]

any two parts the same. Moreover, since the base must be larger than the slope we know that the new base must be the smallest part of λ'. The base of λ' is the slope of λ. Since λ belongs to Class 1, its base is longer than its slope. Thus the slope of λ' is at least as large as the slope of λ, and stricly larger if it meets the base. So λ' must be in Class 2.

Working in the other direction, let λ' belong to Class 2. We can create a new partition, λ by removing the base of λ' and installing it as a new slope. Again we must have a partition with all parts distinct. Suppose the base has size $k.'$ The base is distinct from everything else, and we simply add 1 to the first k rows of λ'. Note that λ lies in Class 1. The slope of λ is equal to the base of λ'. Since the base of λ' is smaller than

its slope, the slope of λ must be smaller than its base.
The nature of these bijection shows that the sizes of Class 1 and Class 2 are equal. The algorithm to go from one to the other is exactly inverted. Also note that the bijection changes the parity of a given partition. From Class 1 to Class 2 we add one part. From Class 2 to Class 1 we remove one part. Thus, in the union of Class 1 and Class 2 the numbers of partitions with even and odd numbers of parts are equal.
Class 3 does not fit into either of these bijections. A partition in Class 3 has the property that the base and slope intersect and either the lengths are equal, or the base is longer than the slope by 1. If the base were 2 unit longer the partition would belong in class 2. Obviously if the slope and base did not intersect the partition would belong to either Class 1 or Class 2. Suppose that λ'' has k parts and belongs to Class 3. Now, imagine we "complete" λ'' in the following way. Add cells to the diagram on the far right side until we obtain a rectangle composed of two squares of area k^2. Figure 16 helps clear up the concept of completion.

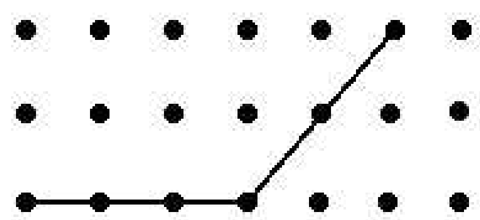

FIGURE 16. The completion of a Class 3 partition

Now we can easily count the number of cells in λ''. There is one section of size k^2 and one section of size
$$\frac{k(k-1)}{2}$$
due to the fact that it is the sum of the integers from 1 to $k-1$. Thus
$$n = k^2 + \frac{k(k-1)}{2} = \frac{3k^2-k}{2} = \frac{k(3k-1)}{2}.$$
If the base exceeds the length of the slope by 1,
$$n = \frac{k(3k+1)}{2}.$$

Thus, if n is not a pentagonal number Class 3 is empty. For some $k \in \mathbb{Z}$ Class 3 will consist of one partition with $|k|$ distinct parts. Thus, if k is even the size of the number of partitions with an even number of parts will be 1 larger. Similarly for odds if k is odd. This proves Euler's pentagonal number theorem. □

9. Extensions

Partitions and tableaux have characteristics that are surprising and intriguing. As with many topics in mathematics, the deeper we explore the more exciting things become. In combinatorics there are a myriad of ways to continue to explore. Symmetric polynomials are the first. I introduce them here as a way to further investigation about partitions.

Symmetric polynomials and combinatorics connect in many different ways. Cameron points out that if a problem does not have an explanation using symmetric polynomials then it is not really a combinatorial problem [1]. Symmetric polynomials can provide a new way to look at some of the problems considered above.

Definition 9.1. *Let $x_1, \ldots, x_N$ be indeterminates. A polynomial $f(x_1, \ldots, x_N)$ is called symmetric if it is left unchanged by any permutation of its arguments* [1].

In other words, if $f(x_{1g}, \ldots, x_{Ng}) = f(x_1, \ldots, x_N)$ for all $g \in S_n$ then f is considered symmetric. In this case S_n is the set of all permutations on $\{1, 2, \ldots, n\}$. There are four special cases of symmetric polynomials that are extremely useful when dealing with combinatorics. Let λ be the partition $n = n_1 + n_2 + \cdots + n_k$.
(1) The *basic polynomial* m_λ is the sum of the terms $x_1^{n_1} \ldots x_k^{n_k}$ and all other terms which can be obtained from this one by permuting the indeterminates.
For example, if λ=1+2+3 then m_λ=

$$\begin{gathered} x_1\ x_2^2\ x_3^3 + \\ x_1^2\ x_2^1\ x_3^3 + \\ x_1^1\ x_2^3\ x_3^2 + \\ x_1^2\ x_2^3\ x_3^1 + \\ x_1^3\ x_2^1\ x_3^2 + \\ x_1^3\ x_2^2\ x_3^1. \end{gathered}$$

(2) The *elemetary symmetric polynomial*, e_n, is the sum of all products of n distinct indeterminates.
For examply, if n is 3 then

$$\begin{aligned} e_1 &= x_1 + x_2 + x_3 \\ e_2 &= x_1x_2 + x_1x_3 + x_2x_3 \\ e_3 &= x_1x_2x_3 \end{aligned}$$

(3) The *complete symmetric polynomial*, h_n, is the sum of all products of n indeterminates (repititions allowed).
For example, if n=3 then

$$\begin{aligned} h_1 &= x_1 + x_2 + x_3 \\ h_2 &= x_1^2 + x_2^2 + x_3^2 + x_1x_2 + x_1x_3 + x_2x_3 \\ h_3 &= x_1^3 + x_2^3 + x_3^3 + x_1^2x_2 + x_1^2x_3 + x_2^2x_1 + x_2^2x_3 + x_3^2x_1 + x_3^2x_2 + x_1x_2x_3 \end{aligned}$$

(4) The *power sum polynomial*, p_n, is equal to $x_1^n + \cdots + x_N^n$.

Suppose that z is one of the symbols e, h or p then $z_\lambda = z_{n_1} \ldots z_{n_k}$ where λ is a partition of n. As usual, an example will greatly help explain these four cases. This example comes from Cameron [1]. If there are three indeterminates, and λ is the partition $3 = 2 + 1$, then

$$\begin{aligned} m_\lambda &= x_1^2x_2 + x_2^2x_1 + x_1^2x_3 + x_3^2x_1 + x_2^2x_3 + x_3^2x_2, \\ e_\lambda &= (x_1x_2 + x_1x_3 + x_2x_3)(x_1 + x_2 + x_3) \\ p_\lambda &= (x_1^2 + x_2^2 + x_3^2)(x_1 + x_2 + x_3) \\ h_\lambda &= e_\lambda + p_\lambda. \end{aligned}$$

These examples do not make the power of symmetric polynomials clear. They can be used to prove things from the generagting function for partition numbers to the forumla for the binomial coefficient. The adaptability of the symmetric polynomials comes from our ability to choose values for the indeterminiates.

As mentioned above, the filling of partition diagrams based on the rules that define standard tableaux is only one possible method. We could stipulate that the cells be filled in a different manner or that the partition be represented in another way. Some other types of tableaux and partition diagrams also have some distinctive combinatorial properties about them as well. In order to define skew diagrams we have to work through two other definitions.

Let $\lambda = a_1 + a_2 + ...$ be a partiton of the integer n. Recall that the partition diagram, $D(\lambda)$ is a graphical representation of the partition λ. The rows of $D(\lambda)$ represent one of the a_is. Also, remember each row is less than or equal to the size of the row above it. As further review, Figure 1 shows the two types of partition diagrams for $\lambda = 4+2+1+1$.

The rank of a given partiton diagram, denoted rank(λ), is the length of the main diagonal of $D(\lambda)$, or equivalently, the largest integer i for which $a_i \geq i$ [4]. Figure 17 shows the rank of a partition digram. Now we can define skew diagrams.

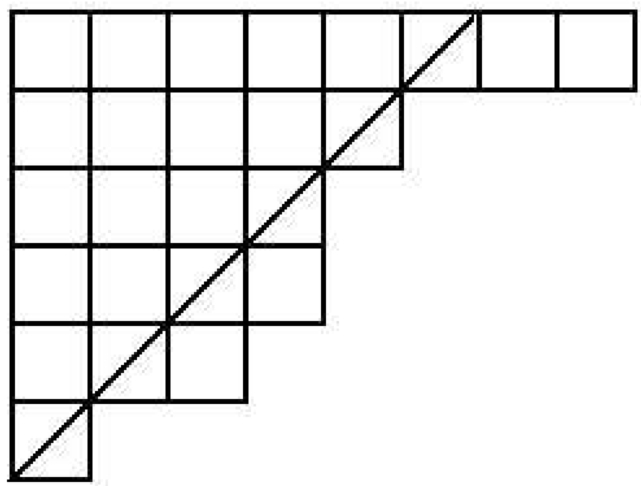

FIGURE 17. The rank of λ=8+5+4+4+3+1

Recall that a standard tableau is a diagram of λ, a partition on n, filled with the integers 1 through n such that the numbers stricly increase down and to the right. A semi-standard tableau works with the same idea on a partition diargram for λ. Let λ_1 be the first row of λ. A semi-standard tableau fills λ with the integers 1 through λ_1 such that they weakly increase to the right and strictly increase down. Regev proves a connection between skew diagrams and semi-standard tableaux.

A skew diagram is effectively the result of subtracting one partition diagram from another. Given two partition diagrams μ and ν, such that ν is completely contained within μ, the the *skew diagram* μ/ν is equal to the complement of $\mu \cap \nu$. Skew diagrams have a few interesting properties about them. They have connections to graphical trees, Schur functions, and of course symmetric polynomials. One of the papers I read deals with the hook numbers of a few diagrams associated with skew diagrams.

Using the hook-length formula, Regev is able to prove a formula for the number of a specific type of skew-diagram for a given n. The result goes much too far into the nature of schur functions to adequately introduce here but the fact that such results exist and provide further study is important to the rest of the work I did.

10. Conclusion

In the above sections I dealt with a variety of seemingly independent topics. From standard tableaux to symmetric polynomials. Each topic comes back to partitions in one way or another. On first glance partitions seem simple. They offer an interesting combinatorial problem, i.e. how many are there, and then we can be done with them. With a bit of creativity partitions open up a wealth of new ideas and problems in combinatorics.

References

[1] Peter J. Cameron. *Combinatorics: Topics, Techniques, Algorithms.* Cambridge University Press, first edition, 2001.

[2] David Feldman. Hook formula.

[3] Amitai Regev and Anatoly Vershik. Asymptotics of Young diagrams and hook numbers. *Electron. J. Combin.*, 4(1):Research Paper 22, 12 pp. (electronic), 1997.

[4] Richard P. Stanley. The rank and minimal border strip decompositions of a skew partition. *J. Combin. Theory Ser. A*, 100(2):349–375, 2002.

[5] A.M. Yaglom and I.M. Yaglom. *Challenging Mathematical Problems with Elementary Solutions: Volume 1.* Dover Publications, first edition, 1964.