**Parabola Volume 62 Issue 2 (2026), Article 1**

# Longest increasing subsequences

**Ryan Bansal**[1]

## 1 Introduction

The longest increasing subsequence (LIS) reveals intrinsic order within data that seem chaotic. Understanding this fundamental structure has profound implications across diverse fields and provides a window into the hidden patterns that govern complex systems.

The study of LIS extends far beyond pure mathematics, serving as a powerful tool for extracting hidden order from random or noisy sequences in many fields. In financial market analysis, LIS algorithms identify sustained periods of growth within volatile price movements, helping detect trends while filtering out short term fluctuations. In bioinformatics and gene sequencing, LIS algorithms are crucial for understanding evolutionary relationships. The LIS length also serves as a "thermometer for disorder" where it can quantitatively measure how organised or chaotic a sequence is. A short LIS indicates a highly jumbled sequence, while a long LIS suggests almost-sorted data. This makes LIS valuable for anomaly detection and data preprocessing in machine learning. More broadly, LIS provides a robust feature for summarising the essential structure of complex sequential data, helping reduce noise while preserving critical ordering information in applications ranging from speech recognition to time series analysis.

This practical need to quantify order within disorder gained urgency in the 1960s as computers began processing increasingly large datasets. Questions about measuring how "sorted" or "random" a sequence is became crucial for algorithm design and data analysis. It was in this context that Stanisław Ulam posed his famous question about the expected behaviour of the longest increasing subsequence in a random permutation. His question was motivated by practical concerns about sorting efficiency and understanding data structure, but it opened up connections to many deeper areas of mathematics. The subsequent work revealed that this simple measure of order connects to the geometry of Young tableaux, representation theory, and even the physics of random matrices. We begin by establishing some definitions and notations, which largely follow from [9, 11].

**Definition 1** (Subsequence)**.** *Let $S_n$ denote the set of permutations of the numbers $1, 2, \dots, n$. If $\sigma \in S_n$, then a* subsequence *of $\sigma$ is a sequence $(\sigma(i_1), \sigma(i_2), \dots, \sigma(i_k))$, where $1 \leq i_1 < i_2 < \cdots < i_k \leq n$. A subsequence is called* increasing *if $\sigma(i_1) < \sigma(i_2) < \cdots < \sigma(i_k)$, and* decreasing *if $\sigma(i_1) > \sigma(i_2) > \cdots > \sigma(i_k)$.*

[1]Ryan Bansal is a student at Lynbrook High School, California.

**Definition 2** (Longest monotone subsequence)**.** *We define $L(\sigma)$ to be the maximal length of an increasing subsequence of the permutation $\sigma$. Similarly, we define $D(\sigma)$ to be the maximal length of a decreasing subsequence of the permutation $\sigma$.*

**Example 3.** *Let $\sigma = (3, 1, 6, 7, 2, 5, 4) \in S_7$. We can verify that $L(\sigma) = 3$ by identifying one increasing subsequence $(3, 6, 7)$ shown in red in Figure 1a while confirming that no increasing subsequence of length 4 exists. Similarly, we can verify that $D(\sigma) = 3$ by examining the decreasing subsequence $(7, 5, 4)$ shown in red in Figure 1b.*

$$\{3, 1, 6, 7, 2, 5, 4\}$$

(a) A longest increasing subsequence

$$\{3, 1, 6, 7, 2, 5, 4\}$$

(b) A longest decreasing subsequence

Figure 1: Examples of longest increasing and decreasing subsequences of $\sigma$

The importance of LIS became more apparent via the Robinson-Schensted correspondence, which asserts that there is a bijection between permutations and pairs of Young tableaux of equal shape. This correspondence reveals that the seemingly simple question of finding LIS is intimately connected to the rich theory of symmetric functions, representation theory, and algebraic combinatorics.

The Robinson-Schensted correspondence not only provides efficient algorithms for computing LIS but also opens the door to understanding the probabilistic behaviour of these structures. Through this lens, we can analyse the expected length of the longest increasing subsequence in random permutations and discover the beautiful limit shape theorem that governs the asymptotic behaviour of Young tableaux.

In this article, we embark on a journey to get a deeper understanding on the theory of LIS. We will see many references to measuring order in chaos. We will use concrete examples, visualisations and algorithms that make the LIS tangible and intuitive before building up to the abstract structures and asymptotic results. We shall see how a simple combinatorial question leads to nontrivial connections across mathematics.

## 2 Patience sorting

We first develop an algorithm called *Patience Sorting* to determine $L(\sigma_n)$ for a given permutation $\sigma_n \in S_n$.

**Definition 4** (Patience Sorting)**.** Given a permutation $\sigma \in S_n$, Patience Sorting builds piles of numbers as follows. For each element $\sigma(i)$,

(a) **New pile**: if the number is larger than all top cards, then start a new pile;

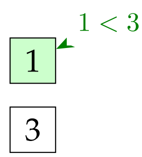
(b) **Bumping down**: otherwise, place the number on the *leftmost* pile whose top card is larger, "bumping down" the rest of the cards in the pile.

**Example 5.** *Take $\sigma = (3, 1, 2, 5, 4) \in S_5$ as the input. We add the sequence elements in order.*

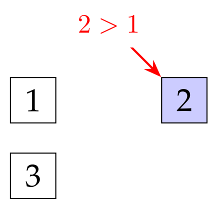
Input: $\boxed{3}$ $\boxed{1}$ $\boxed{2}$ $\boxed{5}$ $\boxed{4}$

*The element 3 is added to a new pile which currently has no element (rule 1).*

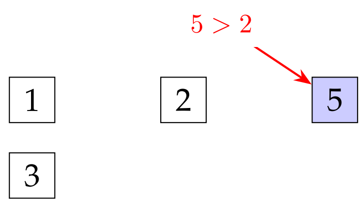

*The element 1 is placed on top of 3 as it is the leftmost pile with a value greater than 1 and 3 is bumped down (rule 2).*



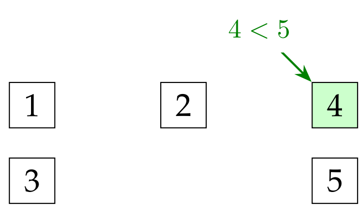

*Since 2 is bigger than the value at the top of all piles (in this case, $2 > 1$), the element 2 is placed on a new pile (rule 1).*




*Since 5 it is greater than the values at the top of all the piles (1 and 2), a new pile with 5 on top is created (rule 1).*




*Finally, 4 is not larger than the top values of all the piles: 4 is greater than 1 and 2, but less than 5. Thus, the leftmost pile with a value larger than 4 is the pile starting with 5 and we place 4 on top of 5 (rule 2).*




*We end up with 3 stacks after applying patience sorting which is the value of $L(\sigma)$. Observe that a new pile is only allowed when the longest increasing subsequence can be extended.*

**Theorem 6** (Schensted [10])**.** *For any permutation $\sigma$, $L(\sigma)$ is the number of stacks at the end of Patience Sorting applied to $\sigma$.*

*Proof.* First, we see that the values at the top of the stacks are in increasing order due to the sorting. This causes elements to always be inserted to keep the top values in order. Additionally, stacks are increasing from top to bottom because of Rule 2, which states that only a value smaller than the current top of a stack can be inserted, pushing the rest of the stack down.

We claim that for an increasing subsequence $(\sigma(i_1), \sigma(i_2), \ldots, \sigma(i_k))$, every element must be in a different stack. Each successive value $\sigma(i_{j+1})$ is greater than the top values for all the stacks that contain $\sigma(i_1), \ldots, \sigma(i_j)$ due to the stacks being increasing, so $\sigma(i_{j+1})$ will be placed in a stack to the right of the stacks containing all the values in the subsequence before it. This bounds the length of an increasing subsequence to at most the amount of stacks.

To obtain an increasing subsequence with exactly the same length as the amount of stacks, start with the top number in the rightmost stack $x = x_s$, and repeatedly go to the value in the stack to the left that was at the top of the stack at the time when $x$ was added to get $x_{s-1}, x_{s-2}, \ldots, x_1$. Due to the order we chose the values in, we also have that $x_1 < x_2 < \cdots < x_s$, so we are done. □

# 3 A fast LIS algorithm

Patience Sorting forms stacks $S_1, S_2, \ldots$ by placing each card $a_k$ on the *leftmost* stack whose top card is at least $a_k$ and creating a new stack if none qualifies. Using this, we see that the index of the stack that receives $a_k$ equals the length of the longest increasing subsequence (LIS) ending at $k$ because all values in later stacks are larger than $a_k$ and cannot be part of the sequence.

Only the top element of each stack influences future placements, so we may compress the stack tops into a single increasing array $L$ and perform a binary search instead of scanning linearly to improve time complexity.

**Algorithm 1** Binary-Search LIS

$L \leftarrow \langle\,\rangle$ ▷ array of current stack tops
**for** $k \leftarrow 1$ **to** $n$ **do**
  **if** $L = \langle\,\rangle$ **or** $a_k > L[|L|]$ **then**
    append $a_k$ to $L$ ▷ new rightmost stack
  **else**
    $i \leftarrow \min\{j : L[j] \geq a_k\}$ ▷ binary search
    $L[i] \leftarrow a_k$ ▷ place $a_k$ atop stack $i$
  **end if**
**end for**
**return** $|L|$ ▷ number of stacks = LIS length

**Theorem 7** (Correctness of Algorithm 1)**.** *The following three conditions are maintained after processing the prefix* $a_1, \dots, a_k$*:*

*(a)* $L[1] < L[2] < \cdots < L[|L|]$*,*

*(b)* $|L| = \ell_k$*, the LIS length of the prefix, and*

*(c) for each* $1 \le r \le |L|$*, the* minimal possible tail *of any increasing subsequence of length* $r$ *inside the prefix is* $L[r]$*.*

*Proof.* Following the initialisation, we split the insertion step into two cases:
**Case 1**: $a_k > L[|L|]$ (appendment).

(a) Appending a larger element preserves strict increase.

(b) A new stack appears, so $|L|$ increases by one. The sequence of stack tops is itself an increasing subsequence of the new length, showing $|L| \ge \ell_k$. Conversely, any subsequence uses at most one element per stack, so $|L| \le \ell_k$ and equality holds.

(c) Tails for lengths $r < |L|$ are unchanged and remain minimal. For $r = |L|$ the unique length-$|L|$ subsequence ends with $a_k$, so $L[|L|] = a_k$ is minimal by definition.

**Case 2**: $a_k \le L[|L|]$ (replacement). Here, the binary search returns the first index $i$ with $L[i] \ge a_k$ and $L[i-1] < a_k \le L[i]$.

(a) Replacing $L[i]$ by the smaller value $a_k$ preserves $L[i-1] < L[i]$ and $L[i] < L[i+1]$ (if they exist), so the array stays strictly increasing.

(b) No new stack is added, so $|L|$ is unchanged. Because we have not created any longer subsequence, $|L| \ge \ell_k$ still holds, while the "one-element-per-stack" argument gives $|L| \le \ell_k$. Hence, $|L| = \ell_k$.

(c) If $r < i$, then the tails are unchanged and are hence minimal. If $r = i$, then the new value $a_k$ is *smaller* than the old tail, and every length-$i$ subsequence ending at or before $k$ has tail $\ge a_k$. Thus, the minimal tail is now $a_k = L[i]$. If $r > i$, then no subsequence of length $r$ uses $L[i]$ as its tail (that slot belonged to shorter subsequences), so their minimal tails are unchanged.

This finishes the proof. □

When $k = n$, condition (b) gives $|L| = L_n$. Each of $n$ iterations performs one binary search in an array of size $\le n$, so the running time is $O(n \log n)$ with space $O(n)$. The "virtual stacks" grow exactly as in patience sorting, just flattened for speed.

We have shown that we can adapt the patience sorting into a fast algorithm to quickly identify the longest increasing subsequence and distil some of the structure in a random permutation. However, we can take it even further by just making a couple tweaks to the algorithm to bring out more information.

# 4 Robinson-Schensted Algorithm

The idea behind the *Robinson-Schensted Algorithm* is to recursively apply patience sorting to each "layer" or "level" of the stacks. Whenever an element is added to a stack, we "bump down" the current top value of the stack to the next layer and apply patience sorting again to add this value to that layer.

**Example 8.** *We illustrate this by looking at an example of recursive patience sorting on the permutation* $(3, 1, 2, 6, 5, 4)$.

$\boxed{3}$ $\boxed{1}$ $\boxed{2}$ $\boxed{6}$ $\boxed{5}$ $\boxed{4}$

*We add* $3$ *to the first layer.*

$\boxed{3}$ Add 3 to layer 1

*Now, we see that* $3$ *is the leftmost number in the first layer. Since* $3 > 1$*, adding* $1$ *bumps* $3$ *down into the second layer. Here,* $3$ *is the largest number and is simply added to the end of the layer.*

$\boxed{1}$ 1 bumps 3

$\boxed{3}$ Add 3 to layer 2

*When adding* $2$*, it is larger than all numbers in the first layer (namely* $1$*), so it is simply added to the end of the layer.*

$\boxed{1}$ $\boxed{2}$ Add 2 to layer 1

$\boxed{3}$

*Similarly,* $6$ *is larger than all current numbers in the first layer, so it is added to the end of the layer.*

$\boxed{1}$ $\boxed{2}$ $\boxed{6}$ Add 6 to layer 1

$\boxed{3}$

*Because* $6$ *is the leftmost number in the first layer greater than* $5$*, it is bumped down and replaced by* $5$*. Now,* $6$ *is in the second layer and it is the largest number, so it is simply added at the end.*

$\boxed{1}$ $\boxed{2}$ $\boxed{5}$ 5 bumps down 6

$\boxed{3}$ $\boxed{6}$ 6 added to layer 3

*Finally, adding $4$ leads to a cascade of bumping operations as seen below. In the first layer, $5$ is the leftmost number greater than $4$, so it is bumped down. In the second layer, the leftmost number greater than $5$ is $6$. Therefore, $5$ bumps $6$ down to the third layer where it settles.*

$\boxed{1}$ $\boxed{2}$ $\boxed{4}$ 4 bumps down 5

$\boxed{3}$ $\boxed{5}$ 5 bumps down 6

$\boxed{6}$ 6 settles

*This gives the final arrangement after applying the algorithm to $(3, 1, 2, 6, 5, 4)$.*

The above example illustrates that many monotonic sequences can be observed in the resulting structure which resembles a *Young tableau*.

# 5 Young tableaux

**Definition 9** (Young diagram). *A* Young diagram *of size $n$ is a collection of boxes across multiple rows, arranged in such a way that the rows are left-justified, and going down, the length of each row is non-increasing. The row lengths form a sequence of non-increasing integers $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ such that $n = |\lambda| = \sum \lambda_i$. Since $\lambda$ gives rise to a partition of $n$ into non-negative integers, we also write $\lambda \vdash n$ and say that the Young diagram is of* shape *$\lambda$.*

**Definition 10** (Young tableau). *A* Young tableau *is a Young diagram filled with integers $1, \ldots, n$ such that each row is increasing from left to right, and each column is increasing from top to bottom.*

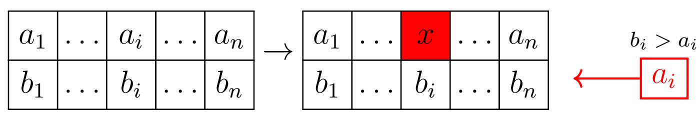

Figure 2: Examples of a Young diagram (left) and Young tableau (right) of size $11$

Let $P$ be the stacking arrangement of numbers found by applying the Robinson-Schensted Algorithm to any permutation $\sigma \in S_n$. In Example 8, $P$ is found in Figure **??**. To add an element $x$, we look at the leftmost position in layer 1 containing an element $y > x$, then replace $y$ with $x$ and repeat the algorithm for the next layer. If $x$ is greater than the entire row, then we simply place it at the end of the row.

**Lemma 11** (Schensted [10]). *Let $P$ be the stacking arrangement of numbers found by applying the Robinson-Schensted Algorithm to any permutation $\sigma \in S_n$. Then $P$ is a Young tableau of size $n$.*

*Proof.* We prove this by induction. The base case $n = 1$ is trivial. We claim that, if we have a Young tableau, then adding a number using recursive patience sorting will preserve the Young tableau structure.

If two adjacent layers $a$ and $b$ have equal length, and an element $x$ is popped from layer $a$ at index $i$, then it must be smaller than the element in layer $b$ at index $i$, so it will be inserted into some index $j \leq i$ in layer $b$. Thus, a row cannot be made longer than the row above it and $P$ always maintain a valid shape of a Young diagram. We now only need to prove that the numbers in each row and column form increasing sequences.




A number is always inserted in a row such that a number to its left is smaller and the number to its right is larger. So, the increasing nature of the rows is always maintained.

When a number $a$ is bumped from one row to the next, it always ends up in a position in the row below which is either directly below its original position, or further to the left. So, $a$ is either directly under the number $b < a$ which displaced it, or a number to the left of $b$ which is less than $b$ and consequently less than $a$. Hence, the insertion method also maintains the increasing nature of the columns.

So, we have shown that the insertion method for recursive patience sorting maintains the increasing nature of the rows and columns. Thus, combined with the base case we have proven that $P$ is always a Young tableau. □

To complete the setup for the Robinson-Schensted correspondence, we construct another tableau $Q$ by recording the order of when spots in the tableau were added. For example, in step 5 of Example 8, 6 is pushed down into the second index of the second layer. Thus, the cell at row 2 column 2 of $Q$ is 5, which indicates that this is the 5th cell to be assigned a number. We can think of $Q$ as the "recording" tableau which keeps track of when moves took place.

**Lemma 12** ([10])**.** *The tableau $Q$ defined above is a Young tableau.*

*Proof.* To insert a value into $P$, we start at the first layer and either replace a value and pop it to the next layer, which does not add any new cells to the layer (only the values in the layer), or adds the value to the end of the layer, which adds a new cell and also terminates the process. So, there is always exactly one new cell made every insertion and it is at the end of a layer.

Going back to $Q$, we see that it has the same shape as $P$ because it is simply recording the order of when each cell in $P$ was filled. Additionally, each digit added to $Q$ is always larger than all the previous digits, and in particular larger than all the digits above it and to its left. So, the numbers in each row and column will always form increasing sequences. □

The motivation behind the creation of $P$ and $Q$ to set up the Robinson-Schensted correspondence is to recreate the original permutation $\sigma$ by "deletion steps", where the information in $Q$ is used in conjunction with the information in $P$ to remove one element at a time from $P$, reconstructing the sequence.

**Lemma 13** (Schensted [10]). *A sequence is uniquely determined by any pair of Young tableaux of the same shape.*

*Proof.* The position of the largest element in $Q$ tells us which number was added onto a row of $P$. This number, call it $x$, must have been displaced from the previous row by the largest number which is smaller than it. There will always be at least one number smaller than $x$ in the previous row because the one directly above $x$ is smaller. The number in the previous row must have also been displaced from the next row up, and so on. In the end, we get to the first row and find what number was inserted into it. We also know what $P$ and $Q$ were before the digit was inserted. This "deletion" step reverses the process of inserting a digit. We can repeat this procedure to find every digit of the sequence, sometimes called the *inverse Robinson-Schensted Algorithm*. $\square$

Lemmas 11 and 12 show that a sequence $\sigma$ uniquely determines two Young tableaux $P$ and $Q$, while Lemma 13 works in the other direction: any two Young tableaux of the same shape uniquely determine a sequence. Thus, we have completed the proof for the Robinson-Schensted correspondence.

**Theorem 14** (Robinson–Schensted correspondence [10]). *The Robinson-Schensted Algorithm creates a mapping that takes a permutation $\sigma \in S_n$ and returns a triple $(\lambda, P, Q)$, where $\lambda \vdash n$, and $P, Q$ are two Young tableaux of shape $\lambda$. This creates a bijection between $S_n$ and ordered pairs of Young tableaux of size $n$ and the same shape.*

Similar to Theorem 6, we also have the following result related to LIS.

**Theorem 15** (Schensted's theorem [10]). *$L(\sigma)$ is the length of the first row of $P$ because it is the same as the number of stacks in patience sorting.*

# 6 Greene's Theorem

The Robinson-Schensted correspondence reveals that the first row of the insertion tableau $P$ captures the length of the longest increasing subsequence. However, the tableau contains far more information in its subsequent rows. A natural question to ask would be to find what combinatorial properties of the permutation do these additional rows encode.

Curtis Greene answered this question in 1974 [4] by showing that the entire shape of the tableau captures information about *multiple* disjoint increasing subsequences. His theorem reveals a beautiful duality between increasing and decreasing subsequences, unified through the structure of Young tableaux.

**Definition 16** (Union of Disjoint Subsequences). *For a permutation $\sigma \in S_n$ and an integer $k \geq 1$, define*

$$L_k(\sigma) \;=\; \max\bigl\{|Y_1 \cup \cdots \cup Y_k| \mid Y_i \text{ is increasing and } Y_i \cap Y_j = \varnothing \text{ for } i \neq j\bigr\},$$

*the largest number of indices that can be covered by $k$ pairwise disjoint increasing subsequences of $\sigma$. Similarly, define $D_k(\sigma)$ for decreasing subsequences.*

Note that $L_1(\sigma) = L(\sigma)$ recovers our original definition of the longest increasing subsequence. The quantity $L_2(\sigma)$ denotes the number of elements that can be covered using two non-overlapping increasing subsequences, and so on.

**Theorem 17** (Greene [4]). *Let $\sigma \in S_n$ and let $\lambda = (\lambda_1, \ldots, \lambda_m)$ be the shape of the insertion tableau $P$ under the Robinson-Schensted correspondence. Then for every $k \geq 1$,*

$$L_k(\sigma) \;=\; \lambda_1 + \lambda_2 + \cdots + \lambda_k \,.$$

*Proof sketch.* A key property of the Robinson-Schensted correspondence is that entries in the same column of $P$ have decreasing indices in the original permutation. More precisely, if $x$ lies above $y$ in column $j$ of $P$, then $x$ appears *after* $y$ in $\sigma$. This means any increasing subsequence can contain at most one entry from each column.

Let $\lambda'_j$ denote the height of the $j$th column. Given $k$ disjoint increasing subsequences, each can contribute at most one element from column $j$, so their union contains at most $\min(k, \lambda'_j)$ elements from that column. Summing over all columns, we obtain

$$L_k(\sigma) \;\leq\; \sum_j \min\bigl(k, \lambda'_j\bigr) \;=\; \lambda_1 + \cdots + \lambda_k \,,$$

where the last equality follows from counting boxes in the first $k$ rows of the Ferrers diagram.

To show that $L_k(\sigma) \geq \lambda_1 + \lambda_2 + \cdots + \lambda_k$, one must construct $k$ disjoint increasing subsequences whose union has size $\lambda_1 + \cdots + \lambda_k$. Greene proves this using the theory of partially ordered sets, showing that the entries in the first $k$ rows of $P$ can always be partitioned into $k$ chains, each forming an increasing subsequence of $\sigma$. □

Greene's theorem becomes even more elegant when reformulated in the language of partially ordered sets. This perspective reveals a deep duality between increasing and decreasing subsequences.

**Definition 18** (Permutation Poset). *For a permutation $\sigma \in S_n$, the* permutation poset $P_\sigma$ *consists of the elements $(i, \sigma(i)), 1 \leq i \leq n$ with the ordering $\leq$ defined by*

$$(i, \sigma(i)) \leq (j, \sigma(j)) \quad\Longleftrightarrow\quad i \leq j \text{ and } \sigma(i) \leq \sigma(j) \,.$$

The structure of $P_\sigma$ captures both increasing and decreasing subsequences. A *chain* in $P_\sigma$ (a totally ordered subset) corresponds to an increasing subsequence of $\sigma$, whereas an *antichain* in $P_\sigma$ (a set with no comparable pairs) corresponds to a decreasing subsequence of $\sigma$.

For a general finite poset $P$, define $c_k(P)$ by the maximal size of a union of $k$ chains and $a_k(P)$ by the maximal size of a union of $k$ antichains. From these, Green constructs two partitions by taking successive differences:

$$\lambda_k(P) = c_k(P) - c_{k-1}(P) \,, \qquad \mu_k(P) = a_k(P) - a_{k-1}(P) \,,$$

with $c_0(P) = a_0(P) = 0$. Below is Greene's famous poset generalisation of Theorem 17. It is often called *Greene's Theorem*[2] but it was also discovered and proved independently by Sergey Fomin in 1978; see [2, 3, 5].

**Theorem 19** (The (Fomin-Greene) Duality Theorem [3, 5]**).**
*The partitions $\lambda(P) = (\lambda_1, \lambda_2, \ldots)$ and $\mu(P) = (\mu_1, \mu_2, \ldots)$ are conjugate partitions of $|P|$.*

For the permutation poset $P_\sigma$, we have $c_k(P_\sigma) = L_k(\sigma)$ and $a_k(P_\sigma) = D_k(\sigma)$. The Duality Theorem then yields the symmetric formula

$$D_k(\sigma) \;=\; \lambda'_1 + \lambda'_2 + \cdots + \lambda'_k\,,$$

where $\lambda'$ denotes the conjugate partition (column lengths of $\lambda$).

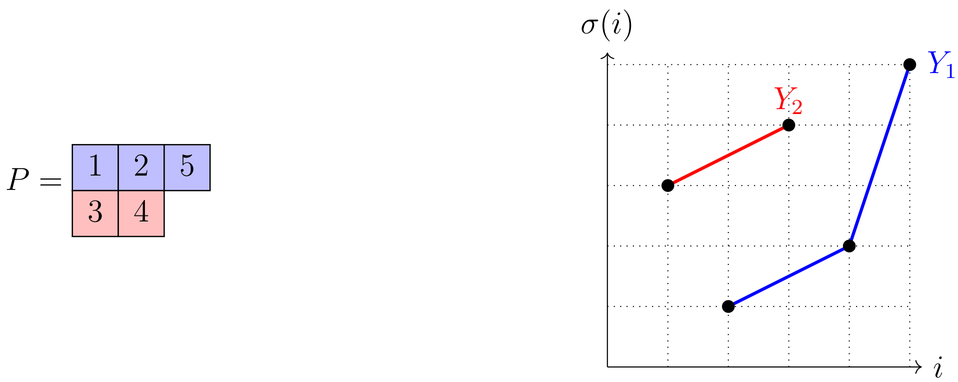



Figure 3: The insertion tableau of $\sigma$ (left) and a dot diagram of $\sigma$ (right)

This symmetry means that the row sums of $\lambda$ control coverings by increasing subsequences, while the column sums control coverings by decreasing subsequences. In particular,

$$L(\sigma) = \lambda_1 \qquad \text{and} \qquad D(\sigma) = \lambda'_1\,.$$

This recovers both the classical Schensted theorem and its decreasing analogue from a single unified framework.

# 7 The hook length

One corollary of Theorem 14 is the following combinatorial identity. For all $\lambda \vdash n$, let $d_\lambda$ be the number of Young tableau of the shape $\lambda$. We have that

$$\sum_{\lambda \vdash n} d_\lambda^2 = n!\,.$$

[2]During 1974–1976, Greene published three famous theorems which are each known in their respective fields as *Greene's Theorem*.

This implies that the probability of a random permutation having a Young tableau of shape $\lambda$ after Robinson-Schensted is

$$\mathbb{P}(\lambda^{(n)} = \lambda) = \frac{d_\lambda^2}{n!}\,.$$

**Definition 20** (Plancherel measure)**.** *The* Plancherel measure *tells us how likely each tableau shape is when we pick a random permutation. Formally, it is the probability measure on partitions of $n$ that assigns probability $\frac{d_\lambda^2}{n!}$ to any partition $\lambda \vdash n$, where $d_\lambda$ is the number of standard Young tableaux of shape $\lambda$.*

Now that we have a probability measure on partitions of $n$, a sensible next question is to calculate the values of $d_\lambda$ for a general Young diagram.

**Definition 21** (Hook length)**.** *The* hook-length *of a cell measures how "central" that cell is in the Young diagram. Specifically, the hook-length $h_\lambda(i,j)$ is the number of cells in the hook $H_\lambda(i,j)$, which consists of cells $(i',j')$ where $i'=i$ and $j' \geq j$, or $j'=j$ and $i' \geq i$.*

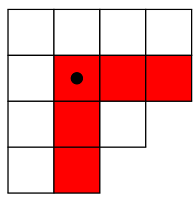


Figure 4: All $H_\lambda(2,2) = 5$ cells for the Young diagram of shape $\lambda = (4,4,3,2)$

Now, we derive a formula for $d_\lambda$ using a probabilistic proof outlined in [6].

**Theorem 22** (Hook length formula [6])**.**

$$d_\lambda = \frac{n!}{\prod_{(i,j)} h_\lambda(i,j)}\,. \tag{1}$$

First, define the "corners" of the Young tableau to be cells which are the last in their row and column. An example is shown below in Figure 5.

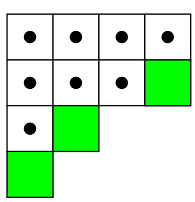


Figure 5: Corners of a Young tableau in green

We use the notation $\mu \nearrow \lambda$ to mean that $\mu$ can be obtained by removing a corner cell of $\lambda$. Since the largest element $n$ always appears in a corner cell of $\lambda$, we have the recurrence

$$d_\lambda = \sum_{\mu \nearrow \lambda} d_\mu \tag{2}$$

and $d_\emptyset = 1$. We now show the right-hand side of (1), which we will denote as $e_\lambda$, satisfies the same recurrence.

Rewrite the recurrence (2) as the sum of probabilities, via

$$\sum_{\mu \nearrow \lambda} \frac{e_\mu}{e_\lambda} = 1\,.$$

Therefore, it suffices to prove that the quantity $e_\mu/e_\lambda$ defines a probability measure on the set of Young diagram $\mu$ with $\mu \nearrow \lambda$. To do this, we define a process called the *hook walk* as illustrated in Figure 6.

**Definition 23** (Hook walk)**.** *The hook walk is a random walk on the cells of a Young diagram $\lambda$ by the following rule:*

*(a) choose a cell $(i,j)$ in $\lambda$ uniformly at random,*

*(b) repeatedly choose successor cell to the current cell in the hook $H_\lambda(i,j)\backslash\{(i,j)\}$,*

*(c) continue until a corner cell $\mathbf{c}$ is selected.*

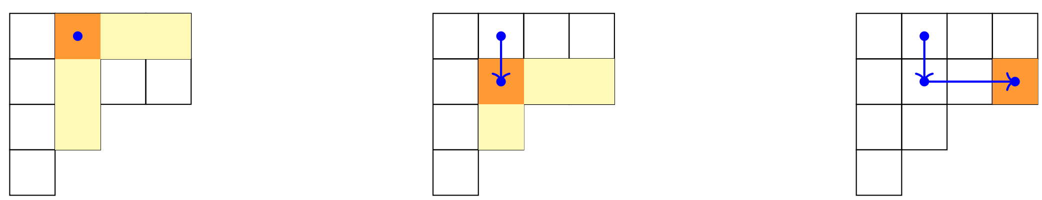


Figure 6: A hook walk in the Young diagram $(4,4,2,1)$ ending at $\mathbf{c} = (2,4)$

**Lemma 24** (Greene [6])**.** *For any corner cell $(a,b)$ of $\lambda$, we have that*

$$\mathbb{P}(\mathbf{c} = (a,b)) = \frac{e_\mu}{e_\lambda}$$

*where $\mu = \lambda\backslash\{(a,b)\}$.*

*Proof.* First, observe that

$$\begin{aligned}\frac{e_\mu}{e_\lambda} &= \frac{1}{n} \prod_{1\le i<a} \frac{h_\lambda(i,b)}{h_\lambda(i,b)-1} \prod_{1\le j<b} \frac{h_\lambda(a,j)}{h_\lambda(a,j)-1} \\ &= \frac{1}{n} \prod_{1\le i<a} \left(1+\frac{1}{h_\lambda(i,b)-1}\right) \prod_{1\le j<b} \left(1+\frac{1}{h_\lambda(a,j)-1}\right). \end{aligned} \tag{3}$$

Now, for a hook walk $(x,y) = (x_1,y_1) \to \cdots \to (x_k,y_k) = (a,b)$, we define the vertical and horizontal projections to be the sets $X = \{x_1, x_2, \dots, x_k\}$ and $Y = \{y_1, y_2, \dots, y_k\}$. We analyse the probability $p(X,Y|x,y)$ that a hook walk starting at $(x,y)$ has the vertical and horizontal projections of $X$ and $Y$. We claim that

$$p(X,Y|x,y) = \prod_{i\in A\backslash\{a\}} \frac{1}{h_\lambda(i,b)-1} \prod_{j\in B\backslash\{b\}} \frac{1}{h_\lambda(a,j)-1}\,.$$

The base case $A = \emptyset$ or $B = \emptyset$ is trivial because there is only one way to move. At each cell, we can either move horizontally to the next $x$-value or the next $y$-value, so we have the recursion

$$p(X, Y \mid x, y) = \frac{1}{h_\lambda(x, y) - 1}(p(X \backslash \{x_1\}, Y \mid x_2, y_1) + p(X, Y \backslash \{y_1\} \mid x_1, y_2)) . \tag{4}$$

By the induction assumption, (4) simplifies to

$$p(X, Y \mid x, y) = \frac{(h_\lambda(x_1, b) - 1 + h_\lambda(a, y_1) - 1)}{h_\lambda(x, y) - 1} \prod_{i \in A \backslash \{a\}} \frac{1}{h_\lambda(i, b) - 1} \prod_{j \in B \backslash \{b\}} \frac{1}{h_\lambda(a, j) - 1} .$$

But $h_\lambda(x, y) - 1 = (h_\lambda(x_1, b) - 1) + (h_\lambda(a, y_1) - 1)$, proving the claim.

Continuing on, the probability $\mathbb{P}(\mathbf{c} = (a, b))$ can be computed by summing the conditional probabilities with respect to the first cell chosen, and then for each such first cell, all possible vertical and horizontal projections. This gives

$$\begin{aligned} \mathbb{P}(\mathbf{c} = (a, b)) &= \frac{1}{n} \sum_{\substack{X \subseteq \{1, \ldots, a\} \\ Y \subseteq \{1, \ldots, b\}}} p(X, Y \mid \min X, \min Y) \\ &= \frac{1}{n} \prod_{1 \leq i < a} \left(1 + \frac{1}{h_\lambda(i, b) - 1}\right) \prod_{1 \leq j < b} \left(1 + \frac{1}{h_\lambda(a, j) - 1}\right), \end{aligned}$$

as desired. □

The hook walk also gives us a way to create a uniformly random Young tableau on a diagram of shape $\lambda$. We simply repeatedly run the hook walk with a random cell each time, and the corner cell it finishes on is filled with numbers starting from $n$ and going down, then removed from the diagram to repeat the algorithm.

# 8 Asymptotics

So far, we have focused on understanding the structure of individual permutations through their longest increasing subsequences and corresponding Young tableaux. Now we turn to a fundamental question about the collective behaviour of all permutations of a given size: what is the expected length of the longest increasing subsequence in a random permutation?

This question, which would later be known as *The Ulam-Hammersley Problem*, was first posed by Stanisław Ulam in 1961 [13]. To understand the nature of this problem, we start with an example illustrating the case $n = 3$.

**Example 25.** *For $n = 3$, there are $3! = 6$ permutations. We tabulate the values $L(\sigma)$ for each $\sigma \in S_3$ below.*

| *Permutation* | *Longest Increasing Subsequence* | *Length* |
|---|---|---|
| $(1,2,3)$ | $(1,2,3)$ | 3 |
| $(1,3,2)$ | $(1,3)$ *or* $(1,2)$ | 2 |
| $(2,1,3)$ | $(1,3)$ *or* $(2,3)$ | 2 |
| $(2,3,1)$ | $(2,3)$ | 2 |
| $(3,1,2)$ | $(1,2)$ | 2 |
| $(3,2,1)$ | *any single element* | 1 |

*The sum of all LIS lengths is $3+2+2+2+2+1 = 12$, so $\mathbb{E}[L_3] = 12/6 = 2$.*

Example 25 illustrates a computational challenge: even for small $n$, determining the expected LIS length requires examining all $n!$ permutations, a task that quickly becomes intractable.

**Definition 26.** *For a uniformly random permutation $\sigma \in S_n$, we define*

$$\ell_n = \mathbb{E}[L(\sigma)] = \frac{1}{n!} \sum_{\sigma \in S_n} L(\sigma)$$

*as the expected length of the longest increasing subsequence.*

John Hammersley's 1972 conjecture [7] provided the first major insight into the asymptotic behaviour of $\ell_n$. Based on computational evidence and heuristic arguments, he conjectured that $\ell_n \sim c\sqrt{n}$ for some constant $c$. This conjecture was remarkable both for its precision and for suggesting a connection to the square root scaling that appears in many probabilistic contexts.

The Robinson-Schensted correspondence reveals that the length of the longest increasing subsequence in a permutation is equal to the length of the first row of its corresponding Young tableau. This connection allows us to reformulate the Ulam-Hammersley Problem in terms of Young tableaux.

Using Theorem 14 and the fact that $L(\sigma) = \lambda_1$, we can express the expected LIS length using the Plancherel measure from Definition 20.

**Lemma 27** (Romik [9])**.** *The expected length of the longest increasing subsequence equals*

$$\ell_n = \sum_{\lambda \vdash n} \lambda_1 \mathbb{P}(\lambda)$$

*where $\mathbb{P}(\lambda) = d_\lambda^2/n!$ is the Plancherel probability.*

To understand $\ell_n$, we need to identify which shapes are most probable. By (1),

$$\log \mathbb{P}(\lambda) = \log n! - \log \left( \prod_{(i,j) \in \lambda} h_\lambda(i,j)^2 \right) = \log n! - 2 \sum_{(i,j) \in \lambda} \log h_\lambda(i,j)\,.$$

Since $\log n!$ is constant for fixed $n$, maximizing $\mathbb{P}(\lambda)$ is equivalent to minimizing

$$S(\lambda) = \sum_{(i,j)\in\lambda} \log h_\lambda(i,j)\,.$$

This sum can be interpreted as the "energy" of a shape. Shapes with lower energy are more probable.

For large $n$, we expect the shape of a typical Young tableau to stabilize in some sense. In the rest of this section, we provide the main idea of the variational approach, which is to view tableaux shapes as continuous objects and to rescale the tableau so that it has a fixed size as $n \to \infty$. See [8, 14] for the more comprehensive technical details.

**Definition 28** (Scaled tableau)**.** *Given a tableau of shape $\lambda$ and size $n$, we scale both axes by $\frac{1}{\sqrt{n}}$, mapping cell $(i,j)$ to $\left(\frac{i}{\sqrt{n}}, \frac{j}{\sqrt{n}}\right)$.*

Under this scaling, each of the $n$ cells has area $\frac{1}{n}$, so the total area remains at 1. To simplify analysis, we rotate by $45$ degrees via the transformation

$$u = \frac{x+y}{\sqrt{2}}\,, \quad v = \frac{y-x}{\sqrt{2}}$$

This transforms the staircase boundary into a smoother curve. In the continuous limit, the discrete energy $S(\lambda)$ converges.

**Lemma 29** (Continuous energy functional [8, 14])**.** *As $n \to \infty$, the energy $S(\lambda)$ converges to*

$$J[\omega] = \int_{-2}^{2}\int_{-2}^{2} \log\left|\frac{\omega(s)-\omega(t)}{s-t}\right| ds\, dt$$

*where $\omega(t)$ describes the rotated boundary.*

The term $\frac{\omega(s)-\omega(t)}{s-t}$ represents the slope between boundary points. Not every $\omega$ corresponds to a valid tableau, so we need a constraint from the tableau property.

**Definition 30** (Lipschitz constraint)**.** *A function $\omega : [-2,2] \to \mathbb{R}$ represents a valid tableau boundary if $|\omega'(t)| \leq 1$ for all $t \in (-2,2)$.*

This constraint ensures rows decrease in length. If the boundary rises too steeply (slope $> 1$), then it would violate the tableau property. We can now find, among all functions $\omega$ satisfying the Lipschitz constraint, the function that maximises $J[\omega]$. This can be done via calculus of variations and the result is presented below without proof.

**Lemma 31** (Optimal shape [8, 14])**.** *The maximizing function has derivative*

$$\omega'(t) = \frac{2}{\pi}\arcsin\left(\frac{t}{2}\right).$$

With Lemma 31 in mind, we can find the explicit form of the limit shape:

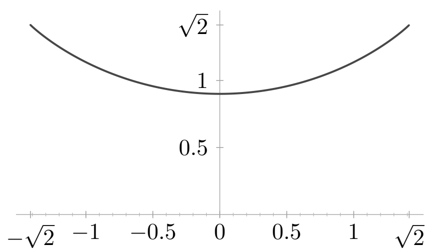
$$\omega(t) = \int_0^t \frac{2}{\pi} \arcsin\left(\frac{s}{2}\right) ds = \frac{2t}{\pi} \arcsin\left(\frac{t}{2}\right) + \frac{2}{\pi}\sqrt{4-t^2} - \frac{4}{\pi} \tag{5}$$

where the last term in (5) is simply a shifting factor that can be ignored. Thus, we finally get an exact form for the limiting shape of a random Young diagram.

**Theorem 32** (Logan-Shepp [8], Vershik-Kerov [14]). *Young diagrams under the Plancherel measure concentrate around the limit shape curve $f_*$, given by*

$$f_*(x) = \frac{2}{\pi}\left(x \arcsin\left(\frac{x}{2}\right) + \sqrt{4-x^2}\right)$$

*for $|x| \le 2$, and $f_*(x) = 0$ for $|x| > 2$.*



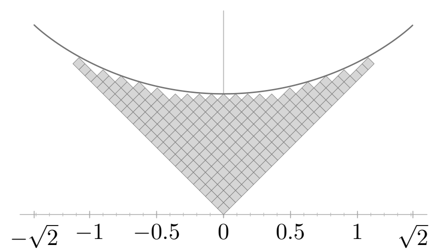




Figure 7: The curve $f_*$ (left) and a Young diagram following the limit shape (right)

The concentration phenomenon means that for large $n$, almost all Young tableaux have shapes very close to this limit curve. The first row length $\lambda_1$ corresponds to the lengths of the diagonal to the limit curve. In the scaled coordinates, this occurs at $(\sqrt{2}, \sqrt{2})$. After scaling back to normal coordinates, this corresponds to a length of $2\sqrt{n}$. This leads us to the following result.

**Theorem 33** (Ulam-Hammersley constant [8, 14]). *For a uniformly random permutation $\sigma$ of length $n$,*

$$\mathbb{E}[L(\sigma)] = 2\sqrt{n} + o(\sqrt{n}) \quad \textit{as } n \to \infty\,.$$

This result shows that random permutations contain a predictable amount of increasing structure. The coefficient $2$ emerges naturally from the geometry of Young tableaux and the solution to the variational problem. The $\sqrt{n}$ scaling reflects the delicate balance between order and disorder in random permutations—enough structure to create substantial increasing subsequences, but not so much as to make the permutation nearly sorted.

# 9 Conclusion

Our journey through the theory of longest increasing subsequences has revealed remarkable connections spanning throughout combinatorics. What began as a simple question about the length of the longest increasing subsequence has led us to results about the asymptotic behaviour of random structures and the geometry of limit shapes.

The Robinson-Schensted correspondence stands as the central pillar of this theory, providing not only efficient algorithms for computing longest increasing subsequences but also a profound connection between the set of permutations and the rich algebraic structure of Young tableaux. Through this correspondence, we see how the seemingly chaotic behaviour of random permutations gives rise to predictable patterns governed by elegant laws.

The theory naturally extends in several fascinating directions. The study of alternating subsequences, patterns that alternate between increasing and decreasing steps, connects longest increasing subsequences to pattern avoidance in permutations and leads to surprising connections with Catalan numbers and Dyck paths. These alternating patterns have found applications in bioinformatics, where they help identify regulatory regions in DNA sequences.

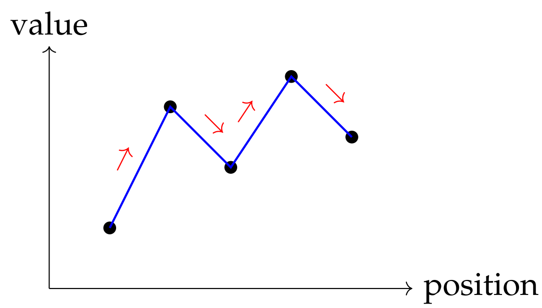



Figure 8: An alternating subsequence showing the up-down pattern that generalises increasing subsequences

Another direction involves the longest common subsequence (LCS) problem, which seeks the longest subsequence shared by multiple sequences. While computing LIS takes $O(n \log n)$ time, finding the LCS for $m$ sequences of length $n$ requires exponential time and becomes NP-hard for multiple sequences. With applications ranging from DNA sequence analysis to plagiarism detection in text, studying this problem is a natural topic for future research.

Perhaps most remarkably, the fluctuations of LIS lengths around their expected value can also be modelled following the Tracy-Widom distribution, originally found in random matrix theory [1, 12]. This connection reveals that the same mathematical structures governing eigenvalue distributions in random matrices also control the fine-scale behaviour of combinatorial objects like permutations. The limit shape theorem itself connects to integrable partial differential equations and variational principles, showing how continuous mathematics emerges from discrete combinatorial problems.

These connections suggest that longest increasing subsequences occupy a special position in the mathematical landscape. They are simple enough to be studied by many people concretely, yet rich enough to reveal deep mathematical truths for those who seek them. The story of longest increasing subsequences is ultimately a story about a simple combinatorial question which can lead to profound insights that illuminate the deep structures underlying seemingly distant topics.

## 10 Acknowledgements

The author thanks Dean Menezes for his guidance and mentorship throughout the process of writing this article. The author would also like to thank Simon Rubinstein-Salzedo for his lectures on paper writing, as well as creating an environment that inspired the author to delve into mathematical writing. This article was written as part of the Euler Circle research program.

## References

[1] J. Baik, P. Deift and K. Johansson, On the distribution of the length of the longest increasing subsequence of random permutations, *Journal of the American Mathematical Society* **12** (1999), 1119–1178.

[2] T. Britz and S. Fomin, Finite posets and Ferrers shapes, *Advances in Mathematics* **158** (2001), 86–127.

[3] S.V. Fomin, Finite partially ordered sets and Young tableaux, *Soviet Mathematics Doklady* **19** (1978), 1510–1514.

[4] C. Greene, An extension of Schensted's theorem, *Advances in Mathematics* **14** (1974), 254–265.

[5] C. Greene, Some partitions associated with a partially ordered set, *Journal of Combinatorial Theory, Series A* **20** (1976), 69–79.

[6] C. Greene, A. Nijenhuis and H.S. Wilf, A probabilistic proof of a formula for the number of Young tableaux of a given shape, *Advances in Mathematics* **31** (1979), 104–109.

[7] J.M. Hammersley, A few seedlings of research, *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Vol. I*, pp. 345–394, University of California Press, 1972.

[8] B.F. Logan and L.A. Shepp, A variational problem for random Young tableaux, *Advances in Mathematics* **26** (1977), 206–222.

[9] D. Romik, *The Surprising Mathematics of Longest Increasing Subsequences*, Cambridge University Press, 2015.

[10] C. Schensted, Longest increasing and decreasing subsequences, *Canadian Journal of Mathematics* **13** (1961), 179–191.

[11] R.P. Stanley, *Enumerative Combinatorics*, Volume 2, Cambridge University Press, 1999.

[12] C.A. Tracy and H. Widom, Level-spacing distributions and the Airy kernel, *Communications in Mathematical Physics* **159** (1994), 151–174.

[13] S.M. Ulam, Monte Carlo calculations in problems of mathematical physics, in *Modern Mathematics for the Engineers*, pp. 261–281, McGraw-Hill, 1961.

[14] A.M. Vershik and S.V. Kerov, Asymptotic behaviour of the Plancherel measure of the symmetric group and the limit form of Young tableaux, *Soviet Mathematics Doklady* **18** (1977), 527–531.