Longest Increasing Subsequences

Ryan Bansal

July 2025

Terminology: Subsequences and LIS

3 1 2 5 4

Subsequence: A sequence obtained by deleting some elements (keeping order)

- **3** 1 **2** **5** 4 (delete 1,4)

Terminology: Subsequences and LIS

3 1 2 5 4

Subsequence: A sequence obtained by deleting some elements (keeping order)

- **3** 1 **2** **5** 4 (delete 1,4)

Increasing subsequence: A subsequence where each element is larger than the previous

- 3 **1** **2** **5** 4 (length 3)
- 3 1 **2** **5** 4 (length 2)
- 3 **1** **2** 5 **4** (length 3)

Longest Increasing Subsequence (LIS): increasing subsequence of maximum length

3 **1** **2** 5 **4** is a LIS because no increasing subsequence has length > 3

Why Study Longest Increasing Subsequences?

LIS is a powerful tool for extracting hidden order from random or noisy sequences.

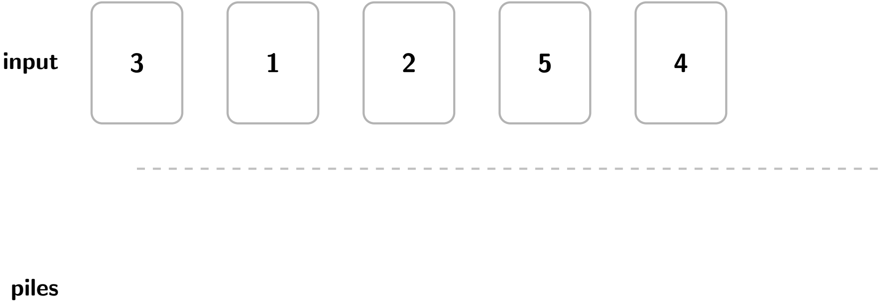
- **Stock market:** Can you spot a long run of rising prices? LIS algorithms help detect trends in noisy financial data.
- **Gene sequencing:** Comparing DNA or protein sequences? LIS finds the longest common ordered patterns—crucial for evolutionary biology.
- **Data cleaning and preprocessing:** The LIS length acts as a **thermometer for disorder**: a short LIS means a jumbled sequence, a long LIS means nearly sorted. Useful for sorting, anomaly detection, and preparing data for analysis.

LIS is also a robust feature for machine learning and pattern recognition, helping to reduce noise and summarize the essential structure of complex data.

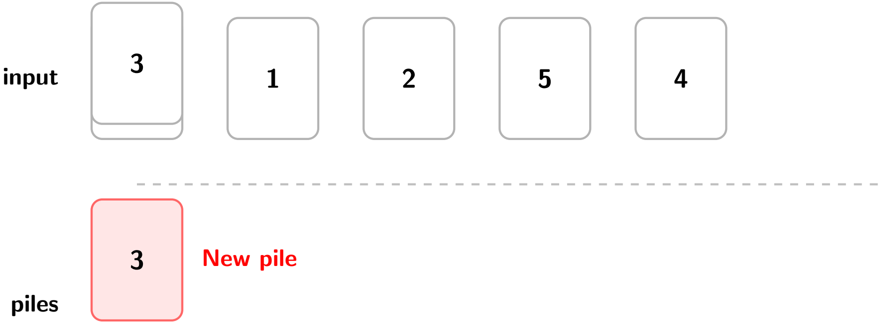
Outline

1. **The Problem**: Longest increasing subsequence in random permutations
2. **Patience Sorting**: A card game that reveals the LIS
3. **Robinson-Schensted Algorithm**: Extracting all ordered layers
4. **Greene's Theorem**: k rows of order
5. **Asymptotics**: The $2\sqrt{n}$ law and limit shapes

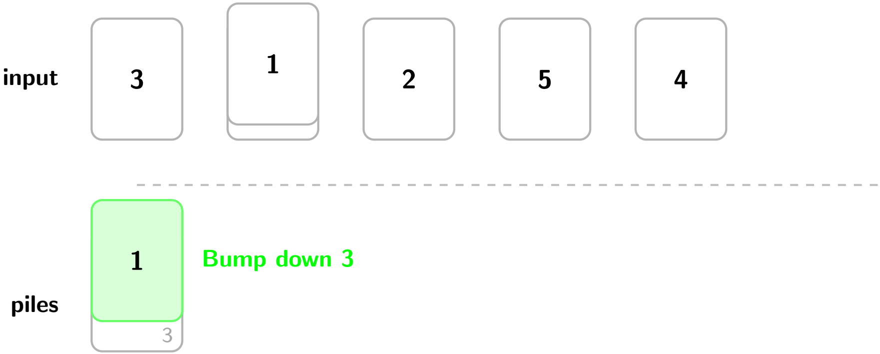
Patience Sorting Demo



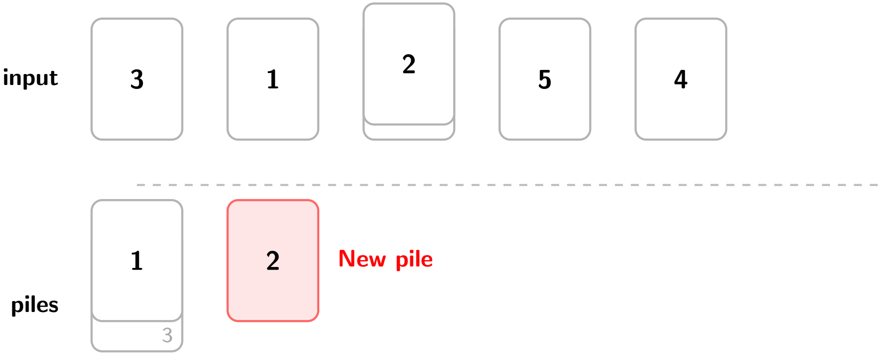
Patience Sorting Demo



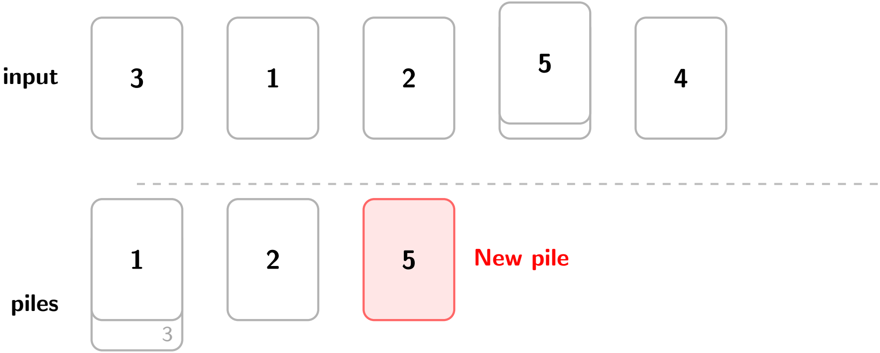
Patience Sorting Demo



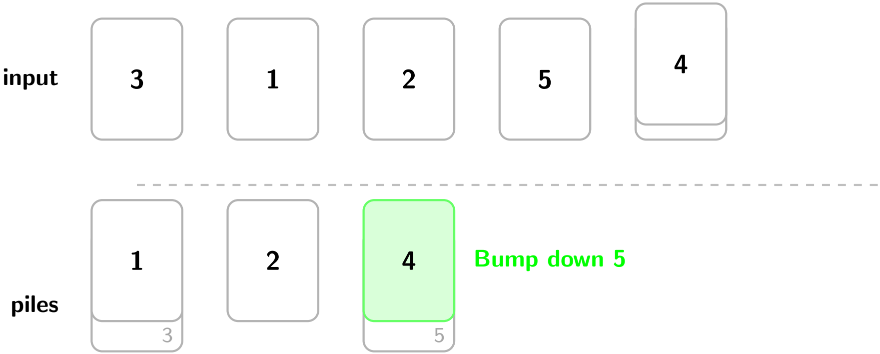
Patience Sorting Demo



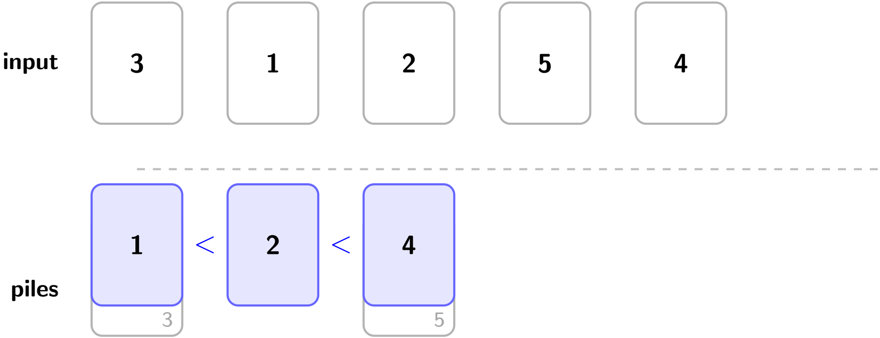
Patience Sorting Demo



Patience Sorting Demo



Patience Sorting Demo



Patience Sorting Rules

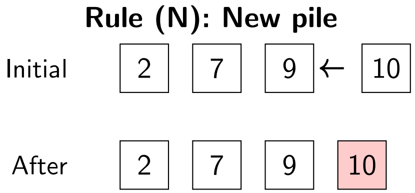
(N) **New pile**: start a new pile if card is larger than all pile tops.

(B) **Bump Down card**: otherwise bump down the left-most top larger than the card.

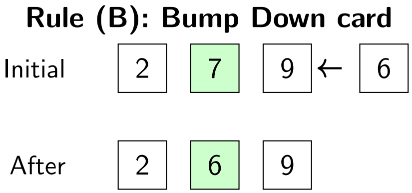
Patience Sorting Rules

(N) **New pile**: start a new pile if card is larger than all pile tops.

(B) **Bump Down card**: otherwise bump down the left-most top larger than the card.



(a) New pile for 10



(b) Bump down 6 onto pile with 7

Patience Sorting Rules

(N) **New pile**: start a new pile if card is larger than all pile tops.

(B) **Bump Down card**: otherwise bump down the left-most top larger than the card.

(a) New pile for 10

(b) Bump down 6 onto pile with 7

Theorem

The amount of piles after patience sorting equals the length of the LIS.

From Patience Sorting to Robinson-Schensted

Patience sorting finds the length of the longest increasing subsequence (LIS), but it wastes the rest of the information in cards which are bumped down.

From Patience Sorting to Robinson-Schensted

Patience sorting finds the length of the longest increasing subsequence (LIS), but it wastes the rest of the information in cards which are bumped down.

The **Robinson–Schensted** extracts *all* the order in the permutation, not just the longest subsequence. Keeps track of numbers in **tableaus:**

- A left-justified array of cells filled with numbers
- Numbers increase along each row and down each column

1	3	5
2	4	

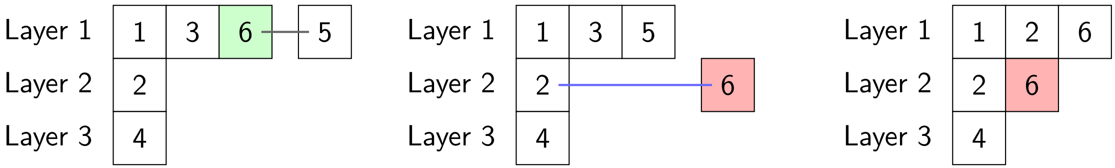
Example of Tableau

Using an *Insertion Tableau*, we keep every value ordered in rows and columns, instead of ignoring them like in patience sorting.
The *Recording Tableau* keeps track of the order in which the tableau is expanded by recording which cell is added per step.

Insertion Rules:

1. If x is larger than whole layer, append x to the end of the layer
2. Otherwise, "bump down" the smallest entry in the layer larger than x

Example: Recursive Bumping for Insertion



(a) Insert 5 into Layer 1 (b) 6 bumps to layer 2 (c) Add 6 to layer 2

Robinson-Schensted Algorithm

input	30	10	20	60	50	40

insertion tableau **recording tableau**

Robinson-Schensted Algorithm

input 30 10 20 60 50 40

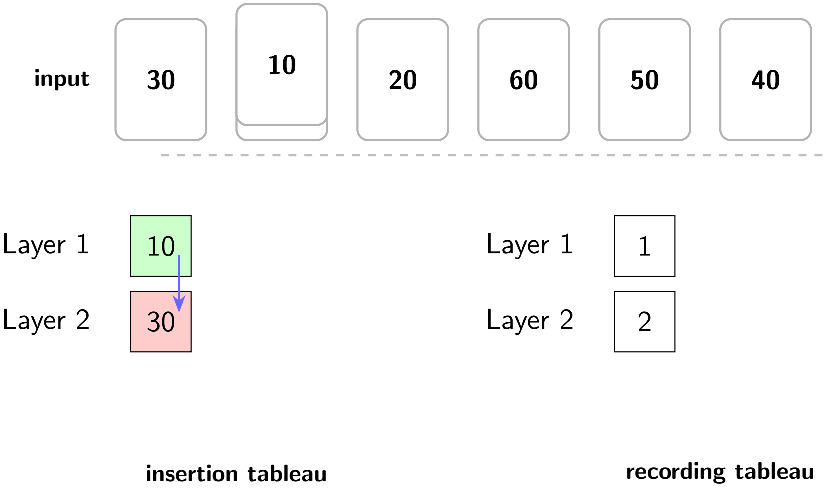
Layer 1 30

Layer 1 1

insertion tableau

recording tableau

Robinson-Schensted Algorithm



Robinson-Schensted Algorithm

input	30	10	20	60	50	40

insertion tableau

Layer 1	10	20
Layer 2	30	

recording tableau

Layer 1	1	3
Layer 2	2	

Robinson-Schensted Algorithm

input					
30	**10**	**20**	**60**	**50**	**40**

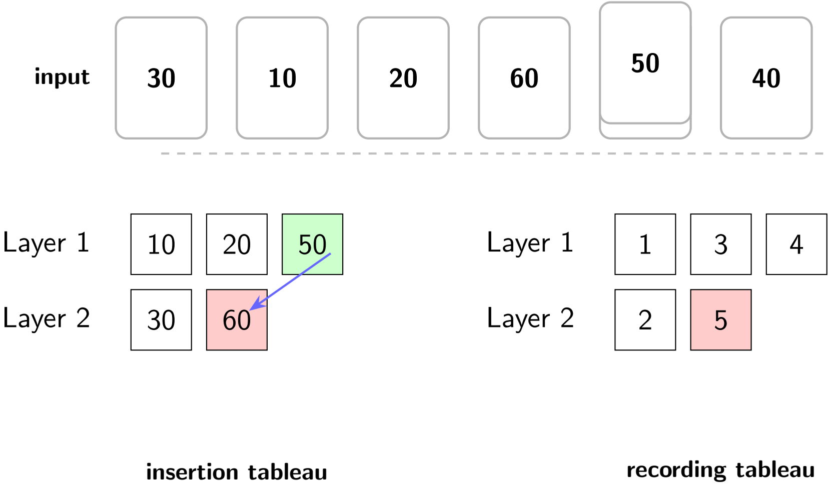
insertion tableau

Layer 1	10	20	60
Layer 2	30		

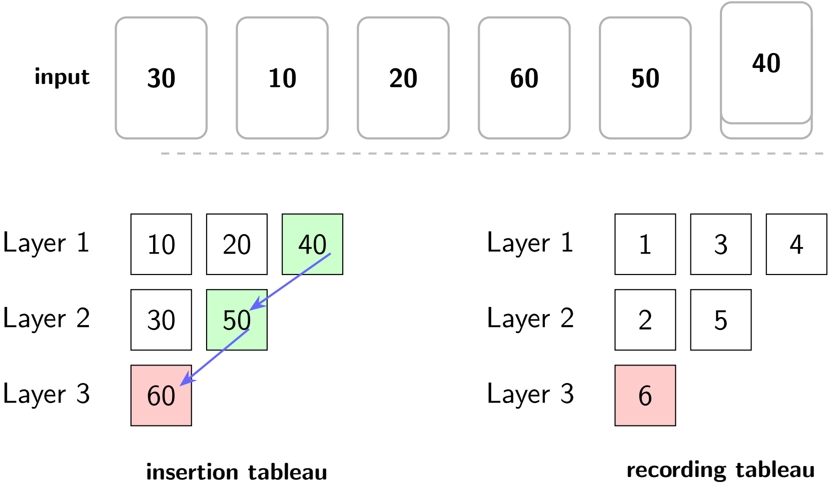
recording tableau

Layer 1	1	3	4
Layer 2	2		

Robinson-Schensted Algorithm



Robinson-Schensted Algorithm



Robinson–Schensted Correspondence

Key insight: Recording tableau makes the process **reversible**
The insertion tableau P stores the values, while the recording tableau Q stores when a cell was added. Together, they encode the complete history of the sequence, allowing us to reconstruct the original sequence by "undoing" each step.

Robinson–Schensted Correspondence

Key insight: Recording tableau makes the process **reversible**

The insertion tableau P stores the values, while the recording tableau Q stores when a cell was added. Together, they encode the complete history of the sequence, allowing us to reconstruct the original sequence by "undoing" each step.

Theorem (Robinson–Schensted Correspondence)

There is a bijection which maps a permutation σ of length n to an ordered pair of Young Tableaux of the same shape.

Robinson–Schensted Correspondence

Key insight: Recording tableau makes the process **reversible**
The insertion tableau P stores the values, while the recording tableau Q stores when a cell was added. Together, they encode the complete history of the sequence, allowing us to reconstruct the original sequence by "undoing" each step.

Theorem (Robinson–Schensted Correspondence)

There is a bijection which maps a permutation σ of length n to an ordered pair of Young Tableaux of the same shape.

Notation:

- We often denote P as the insertion tableau and Q as the recording tableau
- The shape of the tableaus is written $\lambda = (\lambda_1, \lambda_2, \dots)$ where $\lambda_i =$ length of ith layer
- λ is a partition of n

Greene's Theorem

Greene's theorem uses Robinson–Schensted to extract more information from a sequence.

Theorem (Greene's Theorem)

The sum of the first k row lengths in the Robinson-Schensted tableau equals the maximum total size of k disjoint increasing subsequences.

Greene's Theorem

Greene's theorem uses Robinson–Schensted to extract more information from a sequence.

Theorem (Greene's Theorem)

The sum of the first k row lengths in the Robinson-Schensted tableau equals the maximum total size of k disjoint increasing subsequences.

permutation: 5 3 1 2 4 6 7

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

$\lambda_1 = 3$

Remark

This generalizes Schensted's result that $\lambda_1 =$ length of LIS

Greene's Theorem

Greene's theorem uses Robinson–Schensted to extract more information from a sequence.

Theorem (Greene's Theorem)

The sum of the first k row lengths in the Robinson-Schensted tableau equals the maximum total size of k disjoint increasing subsequences.

permutation:	5	3	1	2	4	6	7

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

$\lambda_1 + \lambda_2 = 6$

Remark

The maximum amount of values two disjoint increasing subsequences can cover is 6.

Expected Length

So far, we've focused on the LIS for one permutation at a time. Now, let's look at their general behavior across all permutations of a given size.

Stanisław Ulam (1961): What is the expected length of the longest increasing subsequence in a random permutation of length n?

Expected Length

So far, we've focused on the LIS for one permutation at a time. Now, let's look at their general behavior across all permutations of a given size.

Stanisław Ulam (1961): What is the expected length of the longest increasing subsequence in a random permutation of length n?

For $n = 3$, there are $3! = 6$ permutations:

Permutation	LIS	Length	Permutation	LIS	Length
123	1 2 3	3	231	2 3 1	2
132	1 3 2	2	312	3 1 2	2
213	2 1 3	2	321	3 2 1	1

Sum: $3 + 2 + 2 + 2 + 2 + 1 = 12$ Average: $12/6 = 2$

So $\mathbb{E}[L_3] = 2$.

The Ulam-Hammersley Question

Definition (Ulam's Question)

For a random permutation $\sigma \in S_n$,

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

where $L(\sigma)$ is the length of the longest increasing subsequence in σ.

The Ulam-Hammersley Question

Definition (Ulam's Question)

For a random permutation $\sigma \in S_n$,

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

where $L(\sigma)$ is the length of the longest increasing subsequence in σ.

John Hammersley (1972): Conjectured that $\mathbb{E}[L_n] \sim c\sqrt{n}$ for some constant c.

The Ulam-Hammersley Question

Definition (Ulam's Question)

For a random permutation $\sigma \in S_n$,

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

where $L(\sigma)$ is the length of the longest increasing subsequence in σ.

John Hammersley (1972): Conjectured that $\mathbb{E}[L_n] \sim c\sqrt{n}$ for some constant c.

Theorem (Logan–Shepp (1977), Vershik–Kerov (1977))

For a random permutation of size n,

$$\lim_{n\to\infty} \frac{\mathbb{E}[L_n]}{\sqrt{n}} = 2$$

Limit Shape Theorem

The length of the Longest Increasing Subsequence only represents one layer of the Young tableau. But what about the whole tableau shape as n grows?

Limit Shape Theorem

The length of the Longest Increasing Subsequence only represents one layer of the Young tableau. But what about the whole tableau shape as n grows?

Theorem (Logan–Shepp (1977), Vershik–Kerov (1977))

Sampling from random permutations, as $n \to \infty$, the Young diagram obtained from Robinson-Schensted with shape λ scaled by $\frac{1}{\sqrt{n}}$ and rotated converges almost surely to a curve

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

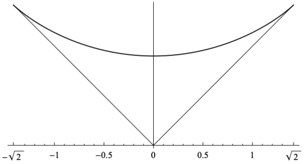
for $-\sqrt{2} \leq x \leq \sqrt{2}$.

Theorem (Logan–Shepp (1977), Vershik–Kerov (1977))

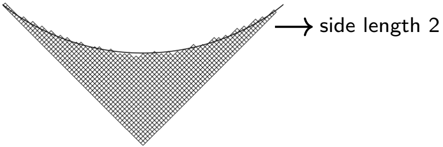
The limit curve of a Young Tableau scaled and rotated is:

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

for $-\sqrt{2} \leq x \leq \sqrt{2}$.



Limit shape curve $\Omega(x)$



Random Young tableau overlayed on the limit shape

Further Work

Alternating patterns:

- Longest alternating subsequence (1, 3, 2 instead of 1, 2, 3)
- Pattern avoidance in permutations
- Applications in bioinformatics for DNA sequences
- Connections to Catalan numbers and Dyck paths

Longest common subsequence:

- Finding common subsequences across multiple sequences
- Fundamental problem in DNA comparism and plagiarism detection
- Proved to be NP-Hard

Promotion:

- Fundamental operation on Young tableaux
- Cycles through different tableaux of same shape
- Applications in representation theory

Thank You!

Questions?