Monotonic Sequence Games

Bruce Sagan
Department of Mathematics
Michigan State University
East Lansing, MI 48824-1027
sagan@math.msu.edu
www.math.msu.edu/~sagan
and
The Otago Theory Group
Department of Computer Science
University of Otago
Dunedin, New Zealand
www.cs.otago.ac.nz

November 6, 2007

The Game on an Interval

The Game on the Rationals

Eine Kleine Game Theory

Open Questions

Outline

Let

$$[n] = \{1, 2, \ldots, n\},$$

and

$$S_n = \text{symmetric group of permutations } \pi = x_1 \ldots x_n \text{ of } [n].$$

Let

$$[n] = \{1, 2, \ldots, n\},$$

and

$$S_n = \text{symmetric group of permutations } \pi = x_1 \ldots x_n \text{ of } [n].$$

A *subsequence of* π *of length* k is $x_{i_1} \ldots x_{i_k}$ with $i_1 < \ldots < i_k$.

Let

$$[n] = \{1, 2, \ldots, n\},$$

and

$$S_n = \text{symmetric group of permutations } \pi = x_1 \ldots x_n \text{ of } [n].$$

A *subsequence of π of length k* is $x_{i_1} \ldots x_{i_k}$ with $i_1 < \ldots < i_k$. The subsequence is *increasing* (respectively, *decreasing*) if $x_{i_1} < \ldots < x_{i_k}$ (respectively, $x_{i_1} > \ldots > x_{i_k}$).

Let

$$[n] = \{1, 2, \ldots, n\},$$

and

$$S_n = \text{symmetric group of permutations } \pi = x_1 \ldots x_n \text{ of } [n].$$

A *subsequence of π of length k* is $x_{i_1} \ldots x_{i_k}$ with $i_1 < \ldots < i_k$. The subsequence is *increasing* (respectively, *decreasing*) if $x_{i_1} < \ldots < x_{i_k}$ (respectively, $x_{i_1} > \ldots > x_{i_k}$). A sequence of either type is *monotone*.

Let

$$[n] = \{1, 2, \ldots, n\},$$

and

$$S_n = \text{symmetric group of permutations } \pi = x_1 \ldots x_n \text{ of } [n].$$

A *subsequence of π of length k* is $x_{i_1} \ldots x_{i_k}$ with $i_1 < \ldots < i_k$. The subsequence is *increasing* (respectively, *decreasing*) if $x_{i_1} < \ldots < x_{i_k}$ (respectively, $x_{i_1} > \ldots > x_{i_k}$). A sequence of either type is *monotone*.

Ex. If $\pi = 6\ 4\ 1\ 3\ 2\ 5$ then $1\ 3\ 5$ is an increasing sequence of length 3 and $6\ 4\ 3$ is a decreasing sequence of length 3.

Let

$$[n] = \{1, 2, \ldots, n\},$$

and

$$S_n = \text{symmetric group of permutations } \pi = x_1 \ldots x_n \text{ of } [n].$$

A *subsequence of π of length k* is $x_{i_1} \ldots x_{i_k}$ with $i_1 < \ldots < i_k$. The subsequence is *increasing* (respectively, *decreasing*) if $x_{i_1} < \ldots < x_{i_k}$ (respectively, $x_{i_1} > \ldots > x_{i_k}$). A sequence of either type is *monotone*.

Ex. If $\pi = 6\ 4\ 1\ 3\ 2\ 5$ then 1 3 5 is an increasing sequence of length 3 and 6 4 3 is a decreasing sequence of length 3.

Theorem (Erdős-Szekeres, 1935)

Any $\pi \in S_{mn+1}$ has either an increasing subsequence of length $m+1$ or a decreasing subsequence of length $n+1$. ■

Let

$$[n] = \{1, 2, \ldots, n\},$$

and

$$S_n = \text{symmetric group of permutations } \pi = x_1 \ldots x_n \text{ of } [n].$$

A *subsequence of π of length k* is $x_{i_1} \ldots x_{i_k}$ with $i_1 < \ldots < i_k$. The subsequence is *increasing* (respectively, *decreasing*) if $x_{i_1} < \ldots < x_{i_k}$ (respectively, $x_{i_1} > \ldots > x_{i_k}$). A sequence of either type is *monotone*.

Ex. If $\pi = 6\ 4\ 1\ 3\ 2\ 5$ then 1 3 5 is an increasing sequence of length 3 and 6 4 3 is a decreasing sequence of length 3.

Theorem (Erdős-Szekeres, 1935)

Any $\pi \in S_{mn+1}$ has either an increasing subsequence of length $m + 1$ or a decreasing subsequence of length $n + 1$. ■

Ex. If $m = 2$ and $n = 3$ then $mn + 1 = 7$. A permutation in S_7, please!!

The Game (Harary-S-West, 1983). Given $m, n \in \mathbb{Z}_{\geq 0}$, players A and B form a sequence $x_1 x_2 \ldots$ of elements of $S = [mn + 1]$ by:

The Game (Harary-S-West, 1983). Given $m, n \in \mathbb{Z}_{\geq 0}$, players A and B form a sequence $x_1 x_2 \ldots$ of elements of $S = [mn + 1]$ by: Player A picks $x_1 \in S$. Then player B picks $x_2 \in S - x_1$, etc.

The Game (Harary-S-West, 1983). Given $m, n \in \mathbb{Z}_{\geq 0}$, players A and B form a sequence $x_1 x_2 \ldots$ of elements of $S = [mn + 1]$ by: Player A picks $x_1 \in S$. Then player B picks $x_2 \in S - x_1$, etc. The first player to form an increasing subsequence of length $m + 1$ or a decreasing subsequence of length $n + 1$ is the winner. By the previous theorem, a winner exists.

The Game (Harary-S-West, 1983). Given $m, n \in \mathbb{Z}_{\geq 0}$, players A and B form a sequence $x_1 x_2 \ldots$ of elements of $S = [mn+1]$ by: Player A picks $x_1 \in S$. Then player B picks $x_2 \in S - x_1$, etc. The first player to form an increasing subsequence of length $m+1$ or a decreasing subsequence of length $n+1$ is the winner. By the previous theorem, a winner exists.

Ex. Let $m = 1$ and $n = 3$ so $[mn+1] = [4]$. Let's play!!

The Game (Harary-S-West, 1983). Given $m, n \in \mathbb{Z}_{\geq 0}$, players A and B form a sequence $x_1 x_2 \ldots$ of elements of $S = [mn+1]$ by: Player A picks $x_1 \in S$. Then player B picks $x_2 \in S - x_1$, etc. The first player to form an increasing subsequence of length $m+1$ or a decreasing subsequence of length $n+1$ is the winner. By the previous theorem, a winner exists.

Ex. Let $m = 1$ and $n = 3$ so $[mn+1] = [4]$. Let's play!!

Note that the game is symmetric in m and n.

The Game (Harary-S-West, 1983). Given $m, n \in \mathbb{Z}_{\geq 0}$, players A and B form a sequence $x_1 x_2 \dots$ of elements of $S = [mn+1]$ by: Player A picks $x_1 \in S$. Then player B picks $x_2 \in S - x_1$, etc. The first player to form an increasing subsequence of length $m+1$ or a decreasing subsequence of length $n+1$ is the winner. By the previous theorem, a winner exists.

Ex. Let $m = 1$ and $n = 3$ so $[mn+1] = [4]$. Let's play!!

Note that the game is symmetric in m and n.

Theorem

The winner of the game on $[mn+1]$ where $m \leq n$ is:

$m \backslash n$	0	1	2	3	4	5	6	7
0	A	A	A	A	A	A	A	A
1		B	A	B	A	B	A	B
2			A	A	A	A	A	A
3				A	A	A	A	?
4					A	?	?	?

where the patterns continue in each of the first 3 rows. ■

Outline

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list* I:

1. Initially $I = \epsilon$, the empty sequence.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \dots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \dots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi =$

$$I: \quad \epsilon,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4$

$$I: \quad \epsilon,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4$

$$I: \quad \epsilon, \quad 4,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2$

$$I: \quad \epsilon, \quad 4,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list* I:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2$

$$I: \quad \epsilon, \quad 4, \qquad 2,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5$

$$I: \quad \epsilon, \quad 4, \qquad 2,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list* I:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5$

$$I: \quad \epsilon, \quad 4, \qquad 2, \quad 2\ 5,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3$

$$I: \quad \epsilon, \quad 4, \quad\quad 2, \quad 2\ 5,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3$

$$I: \quad \epsilon, \quad 4, \qquad 2, \quad 2\ 5, \quad 2\ 3,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list* I:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1$

$$I: \quad \epsilon, \quad 4, \quad\quad 2, \quad 2\ 5, \quad 2\ 3,$$

Play the same game with $[mn + 1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1$

$$I: \quad \epsilon, \quad 4, \quad 2, \quad 2\ 5, \quad 2\ 3, \quad 1\ 3,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$I: \quad \epsilon, \quad 4, \quad 2, \quad 2\ 5, \quad 2\ 3, \quad 1\ 3,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$I: \quad \epsilon, \quad 4, \qquad 2, \quad 2\ 5, \quad 2\ 3, \qquad 1\ 3, \quad 1\ 3\ 6$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$I: \quad \epsilon, \quad 4, \quad 2, \quad 2\ 5, \quad 2\ 3, \quad 1\ 3, \quad 1\ 3\ 6$$

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$I: \quad \epsilon, \quad 4, \quad 2, \quad 2\ 5, \quad 2\ 3, \quad 1\ 3, \quad 1\ 3\ 6$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \textit{length of a longest increasing subsequence ending at } x_i,$$

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \dots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \dots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$I: \quad \epsilon, \quad 4, \quad 2, \quad 2\,5, \quad 2\,3, \quad 1\,3, \quad 1\,3\,6$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{rl} I: & \epsilon,\ \ 4,\ \ \ \ 2,\ \ 2\,5,\ \ 2\,3,\ \ \ \ 1\,3,\ \ 1\,3\,6 \\ D: & \epsilon, \end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{rllllllll} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & & & & & \end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \textit{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list* I:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{llllllll} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & & & & \end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & & & \end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & & \end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{llllllll}
I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\
D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, &
\end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1x_2\ldots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1y_2\ldots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{rccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I,

$$j = \text{length of a longest increasing subsequence ending at } x_i,$$

Similarly build a *decreasing list D* by reversing the inequalities.

Play the same game with $[mn+1]$ replaced by $\mathbb{Q}$. As $\pi = x_1 x_2 \dots$ is built, also build the *increasing list I*:

1. Initially $I = \epsilon$, the empty sequence.
2. If $I = y_1 y_2 \dots$ when x_i is picked, have x_i replace the smallest $y_j > x_i$ or append x_i to the right end of I if no such y_j exists.

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{rccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \end{array}$$

Since 6 was placed in the third column of I we have an increasing subsequence of length three ending at 6, e.g., 2 3 6.

Theorem (Schensted, 1961)

If x_i is placed in column j of I, and in column k of D

$$\begin{aligned} j &= \textit{length of a longest increasing subsequence ending at } x_i, \\ k &= \textit{length of a longest decreasing subsequence ending at } x_i. \end{aligned}$$

Similarly build a *decreasing list D* by reversing the inequalities.

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$x_i \in I \quad \text{and} \quad x_i \notin D \quad \Longrightarrow \quad \text{color } x_i \text{ with } R,$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$
\begin{array}{llllll}
x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\
x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B,
\end{array}
$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} \quad x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{llllllll} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$C: \quad \epsilon,$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llcl} x_i \in I \text{ and } x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I \text{ and } x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I \text{ and } x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \end{array}$$

$$C: \quad \epsilon, \quad \overset{4}{P},$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llcll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B}\ ,$$

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow \quad \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow \quad \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow \quad \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\,2\,5\,3\,1\,6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \end{array}$$

$$C: \quad \epsilon, \quad \overset{4}{P}, \quad \overset{2}{P}\,\overset{4}{B}, \quad \overset{2}{P}\,\overset{5}{P},$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} \quad x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lllllll} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B}\ , \quad \overset{2}{P}\ \overset{5}{P}, \quad \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B},$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll}
x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\
x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\
x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P.
\end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc}
I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\
D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \\
C: & \epsilon, & \overset{4}{P}, & \overset{2}{P}\ \overset{4}{B}\ , & \overset{2}{P}\ \ \overset{5}{P}, & \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B}\ , &
\end{array}$$

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{lllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \\ & & & & & & & \\ C: & \epsilon, & \overset{4}{P}, & \overset{2}{P}\ \overset{4}{B}, & \overset{2}{P}\ \overset{5}{P}, & \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{6}{P} \end{array}$$

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lllllll} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B}, \quad \overset{2}{P}\ \overset{5}{P}, \quad \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, \quad \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B}, \quad \overset{1}{P}\ \overset{3}{P}\ \overset{6}{P}$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llcll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\,2\,5\,3\,1\,6$

$$\begin{array}{lcccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \\ C: & \epsilon, & \overset{4}{P}, & \overset{2}{P}\,\overset{4}{B}, & \overset{2}{P}\,\overset{5}{P}, & \overset{2}{R}\,\overset{3}{P}\,\overset{5}{B}, & \overset{1}{P}\,\overset{3}{P}\,\overset{5}{B}, & \overset{1}{P}\,\overset{3}{P}\,\overset{6}{P} \end{array}$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llcl}
x_i \in I & \text{and} \quad x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\
x_i \notin I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\
x_i \in I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P.
\end{array}$$

Ex. $\pi = 4\,2\,5\,3\,1\,6$

$$\begin{array}{lccccccc}
I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\
D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \\
C: & \epsilon, & \overset{4}{P}, & \overset{2}{P}\ \overset{4}{B}, & \overset{2}{P}\ \overset{5}{P}, & \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{6}{P}
\end{array}$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \\ C: & \epsilon, & \overset{4}{P}, & \overset{2}{P}\ \overset{4}{B}, & \overset{2}{P}\ \overset{5}{P}, & \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{6}{P} \end{array}$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$
\begin{array}{llllll}
x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\
x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\
x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P.
\end{array}
$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$
\begin{array}{llllllll}
I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\
D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \\
C: & \epsilon, & \overset{4}{P}, & \overset{2}{P}\ \overset{4}{B}, & \overset{2}{P}\ \overset{5}{P}, & \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B}, & \overset{1}{P}\ \overset{3}{P}\ \overset{6}{P}
\end{array}
$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

I :	ϵ,	4,	2,	2 5,	2 3,	1 3,	1 3 6
D :	ϵ,	4,	4 2,	5 2,	5 3,	5 3 1,	6 3 1

C : ϵ,

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{lllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{llllllll} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \overset{4}{P},$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll} x_i \in I & \text{and} \quad x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{llllllll} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \end{array}$$

$$C: \quad \epsilon, \quad {\uparrow}\overset{4}{P},$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

I :	ϵ,	4,	2,	2 5,	2 3,	1 3,	1 3 6
D :	ϵ,	4,	4 2,	5 2,	5 3,	5 3 1,	6 3 1

$$C : \quad \epsilon, \quad {}_{\uparrow}\overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B}\ ,$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\,2\,5\,3\,1\,6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\ D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1 \end{array}$$

$$C: \quad \epsilon, \quad \uparrow\overset{4}{P}, \quad \overset{2}{P}\,\overset{4}{B}\uparrow,$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{lllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{llllllll} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \uparrow\overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B}\uparrow, \quad \overset{2}{P}\ \ \overset{5}{P},$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.

Algorithm for C. 1. Initially $C = \epsilon$.

2. Each x_i inserts a P into the corresponding space of C.

3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \uparrow \overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B} \uparrow, \quad \overset{2}{P} \uparrow \overset{5}{P},$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \\ & & & & & & & \\ C: & \epsilon, & \uparrow\overset{4}{P}, & \overset{2}{P}\ \overset{4}{B}\uparrow, & \overset{2}{P}\uparrow\overset{5}{P}, & \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, & & \end{array}$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \uparrow\overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B}\uparrow, \quad \overset{2}{P}\uparrow\overset{5}{P}, \quad \uparrow\overset{2}{R}\ \overset{3}{P}\ \overset{5}{B},$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llll}
x_i \in I & \text{and} \quad x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\
x_i \notin I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\
x_i \in I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P.
\end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc}
I: & \epsilon, & 4, & 2, & 2\,5, & 2\,3, & 1\,3, & 1\,3\,6 \\
D: & \epsilon, & 4, & 4\,2, & 5\,2, & 5\,3, & 5\,3\,1, & 6\,3\,1
\end{array}$$

$$C: \quad \epsilon, \quad \uparrow\overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B}\uparrow, \quad \overset{2}{P}\uparrow\overset{5}{P}, \quad \uparrow\overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, \quad \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B}\ ,$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list C* by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$\begin{array}{llllll} x_i \in I & \text{and} & x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\ x_i \notin I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\ x_i \in I & \text{and} & x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P. \end{array}$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

$$\begin{array}{lccccccc} I: & \epsilon, & 4, & 2, & 2\ 5, & 2\ 3, & 1\ 3, & 1\ 3\ 6 \\ D: & \epsilon, & 4, & 4\ 2, & 5\ 2, & 5\ 3, & 5\ 3\ 1, & 6\ 3\ 1 \end{array}$$

$$C: \quad \epsilon, \quad \uparrow \overset{4}{P}, \quad \overset{2}{P}\ \overset{4}{B} \uparrow, \quad \overset{2}{P} \uparrow \overset{5}{P}, \quad \uparrow \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}, \quad \overset{1}{P}\ \overset{3}{P}\ \overset{5}{B} \uparrow,$$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.
Algorithm for C. 1. Initially $C = \epsilon$.
2. Each x_i inserts a P into the corresponding space of C.
3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Build a *combined list* C by assigning colors red (R), blue (B), and purple (P) to the x_i as follows:

$$
\begin{array}{llll}
x_i \in I & \text{and} \quad x_i \notin D & \Longrightarrow & \text{color } x_i \text{ with } R, \\
x_i \notin I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } B, \\
x_i \in I & \text{and} \quad x_i \in D & \Longrightarrow & \text{color } x_i \text{ with } P.
\end{array}
$$

Ex. $\pi = 4\ 2\ 5\ 3\ 1\ 6$

I :	ϵ,	4,	2,	2 5,	2 3,	1 3,	1 3 6
D :	ϵ,	4,	4 2,	5 2,	5 3,	5 3 1,	6 3 1
C :	ϵ,	$\uparrow \overset{4}{P}$,	$\overset{2}{P}\ \overset{4}{B} \uparrow$,	$\overset{2}{P} \uparrow \overset{5}{P}$,	$\uparrow \overset{2}{R}\ \overset{3}{P}\ \overset{5}{B}$,	$\overset{1}{P}\ \overset{3}{P}\ \overset{5}{B} \uparrow$,	$\overset{1}{P}\ \overset{3}{P}\ \overset{6}{P}$

R and P are called *redish* and *draining red* is $R \leftarrow \epsilon$, $P \leftarrow B$.
B and P are called *bluish* and *draining blue* is $B \leftarrow \epsilon$, $P \leftarrow R$.

Algorithm for C. 1. Initially $C = \epsilon$.

2. Each x_i inserts a P into the corresponding space of C.

3. Drain red from the closest redish element to the right of the new P (if any), and drain blue from the closest bluish element to the left of the new P (if any).

Theorem (Otago-S)

The winner of the game on $\mathbb{Q}$ where $m \leq n$ is:

$m \backslash n$	0	1	2	3	4	5	6	7
0	A	A	A	A	A	A	A	A
1		B	B	B	B	B	B	B
2			A	B	A	B	A	B
3				A	A	A	A	A
4					A	A	A	A

where the patterns continue in each of the first 5 rows. ∎

Outline

Let G be a 2-person, “last player to move wins” game with no draw positions.

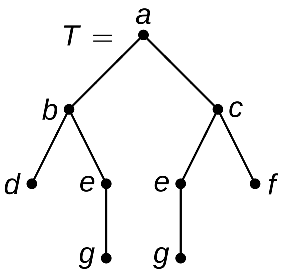
Let G be a 2-person, “last player to move wins” game with no draw positions. The *game tree, T* of G has

nodes v = positions of G,

Let G be a 2-person, “last player to move wins” game with no draw positions. The *game tree, T* of G has

nodes v	=	positions of G,
children of v	=	all positions reachable in one move from v.

Ex.



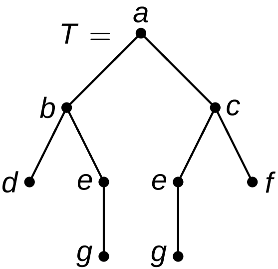
Let G be a 2-person, "last player to move wins" game with no draw positions. The *game tree,* T of G has

nodes v = positions of G,

children of v = all positions reachable in one move from v.

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.



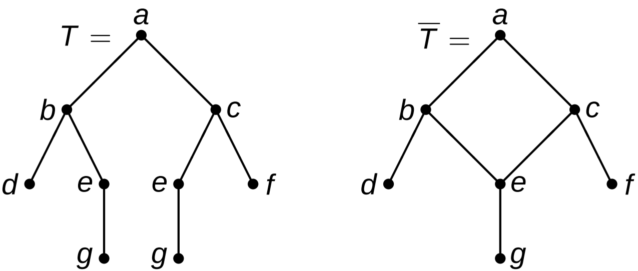
Let G be a 2-person, "last player to move wins" game with no draw positions. The *game tree,* T of G has

nodes v = positions of G,

children of v = all positions reachable in one move from v.

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.



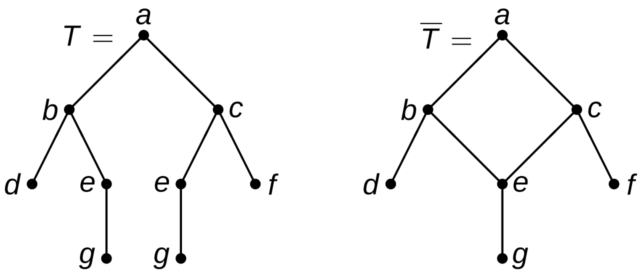
Let G be a 2-person, “last player to move wins” game with no draw positions. The *game tree, T* of G has

nodes v = positions of G,

children of v = all positions reachable in one move from v.

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.

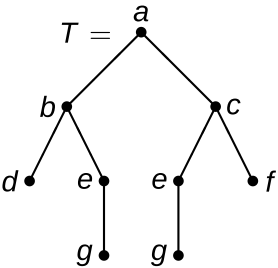


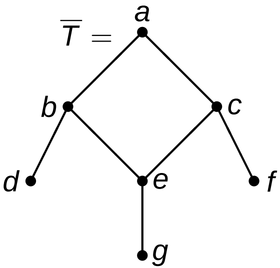
Let $\ell(v) = \begin{cases} \mathcal{N} & \text{if the next player wins from position } v, \\ \mathcal{P} & \text{if the previous player wins from position } v. \end{cases}$

Let G be a 2-person, “last player to move wins” game with no draw positions. The *game tree, T* of G has

nodes v	=	positions of G,
children of v	=	all positions reachable in one move from v.

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex. $T =$ 

$\overline{T} =$ 

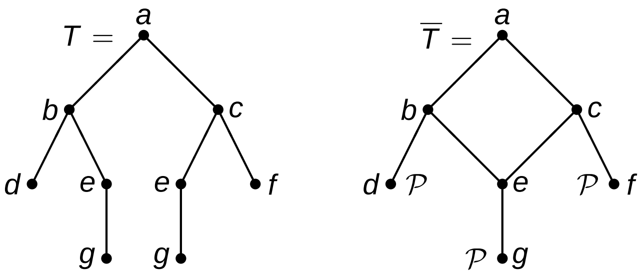
Let $\ell(v) = \begin{cases} \mathcal{N} & \text{if the next player wins from position } v, \\ \mathcal{P} & \text{if the previous player wins from position } v. \end{cases}$

Now label all terminal $v \in \overline{T}$ with $\mathcal{P}$, Let G be a 2-person, “last player to move wins” game with no draw positions. The *game tree*, T of G has

nodes v	=	positions of G,
children of v	=	all positions reachable in one move from v.

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.



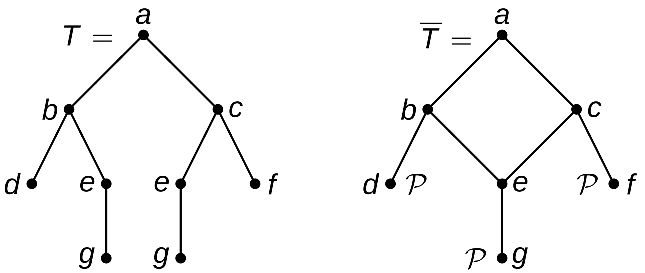
Let $\ell(v) = \begin{cases} \mathcal{N} & \text{if the next player wins from position } v, \\ \mathcal{P} & \text{if the previous player wins from position } v. \end{cases}$

Now label all terminal $v \in \overline{T}$ with $\mathcal{P}$, Let G be a 2-person, "last player to move wins" game with no draw positions. The *game tree, T* of G has

$$\begin{aligned} \text{nodes } v \quad &= \quad \text{positions of } G, \\ \text{children of } v \quad &= \quad \text{all positions reachable in one move from } v. \end{aligned}$$

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.



Let $\ell(v) = \begin{cases} \mathcal{N} & \text{if the next player wins from position } v, \\ \mathcal{P} & \text{if the previous player wins from position } v. \end{cases}$

Now label all terminal $v \in \overline{T}$ with $\mathcal{P}$, and work upwards using:

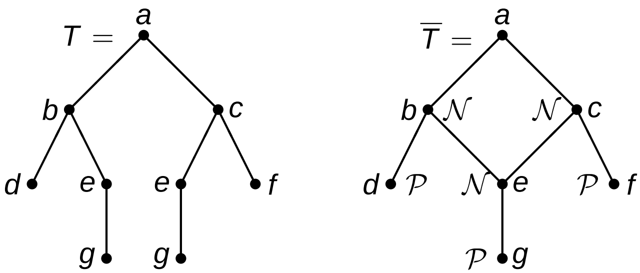
(i) if there is a $\mathcal{P}$-child of v then let $\ell(v) = \mathcal{N}$

Let G be a 2-person, "last player to move wins" game with no draw positions. The *game tree,* T of G has

$$\begin{aligned} \text{nodes } v &= \text{positions of } G, \\ \text{children of } v &= \text{all positions reachable in one move from } v. \end{aligned}$$

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.



Let $\ell(v) = \begin{cases} \mathcal{N} & \text{if the next player wins from position } v, \\ \mathcal{P} & \text{if the previous player wins from position } v. \end{cases}$

Now label all terminal $v \in \overline{T}$ with $\mathcal{P}$, and work upwards using:

(i) if there is a $\mathcal{P}$-child of v then let $\ell(v) = \mathcal{N}$

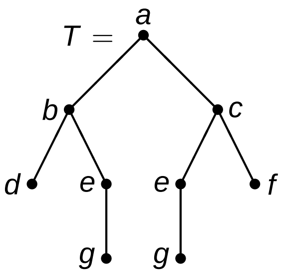
Let G be a 2-person, “last player to move wins” game with no draw positions. The *game tree*, T of G has

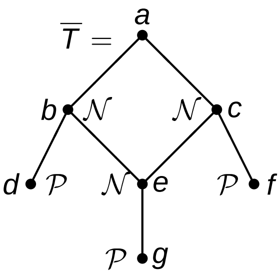
nodes v = positions of G,

children of v = all positions reachable in one move from v.

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.





Let $\ell(v) = \begin{cases} \mathcal{N} & \text{if the next player wins from position } v, \\ \mathcal{P} & \text{if the previous player wins from position } v. \end{cases}$

Now label all terminal $v \in \overline{T}$ with $\mathcal{P}$, and work upwards using:

(i) if there is a $\mathcal{P}$-child of v then let $\ell(v) = \mathcal{N}$

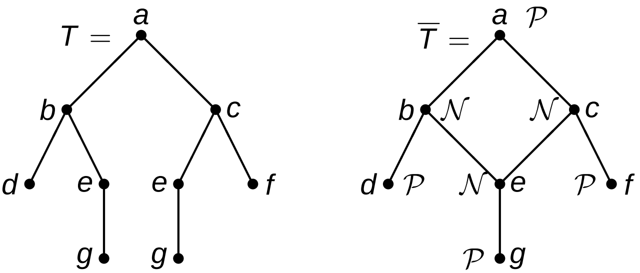
(ii) if all children of v are $\mathcal{N}$ then let $\ell(v) = \mathcal{P}$

Let G be a 2-person, "last player to move wins" game with no draw positions. The *game tree*, T of G has

$$\text{nodes } v \quad = \quad \text{positions of } G,$$
$$\text{children of } v \quad = \quad \text{all positions reachable in one move from } v.$$

If the same position w is a child of both v and v' then identify the copies of w to get a (di)graph $\overline{T}$.

Ex.

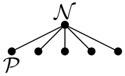


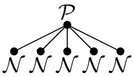
Let $\ell(v) = \begin{cases} \mathcal{N} & \text{if the next player wins from position } v, \\ \mathcal{P} & \text{if the previous player wins from position } v. \end{cases}$

Now label all terminal $v \in \overline{T}$ with $\mathcal{P}$, and work upwards using:

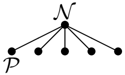
(i) if there is a $\mathcal{P}$-child of v then let $\ell(v) = \mathcal{N}$

(ii) if all children of v are $\mathcal{N}$ then let $\ell(v) = \mathcal{P}$

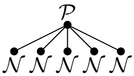
(i) 
$\mathcal{N}$
$\mathcal{P}$

(ii) 
$\mathcal{P}$
$\mathcal{N}$ $\mathcal{N}$ $\mathcal{N}$ $\mathcal{N}$ $\mathcal{N}$

(i)

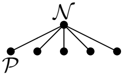


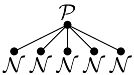
(ii)



Theorem (Otago-S)

The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

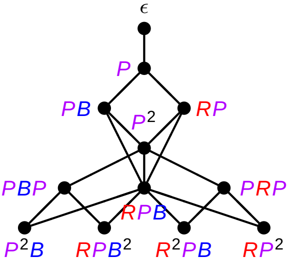
(i) 

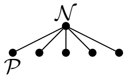
(ii) 

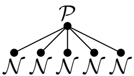
Theorem (Otago-S)

The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

Proof. Part of the top of $\overline{T}$ is



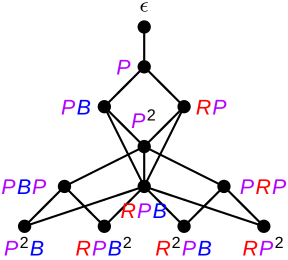
(i) 

(ii) 

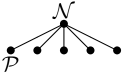
Theorem (Otago-S)

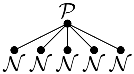
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

Proof. Part of the top of $\overline{T}$ is



Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$.

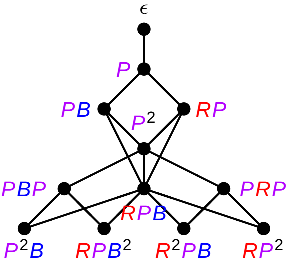
(i) 

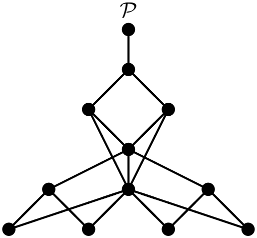
(ii) 

Theorem (Otago-S)

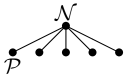
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

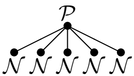
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$.

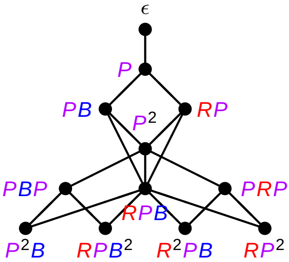
(i) 

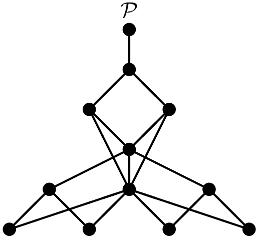
(ii) 

Theorem (Otago-S)

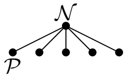
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

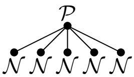
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii).

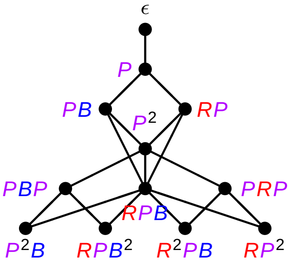
(i) 

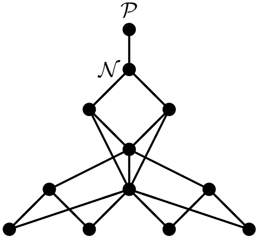
(ii) 

Theorem (Otago-S)

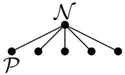
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

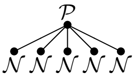
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii).

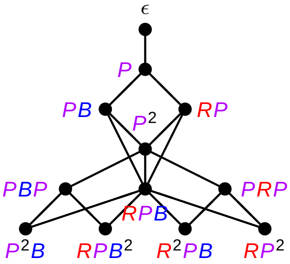
(i) 

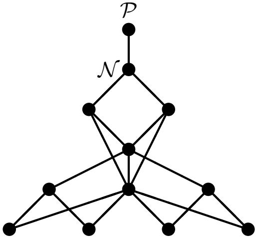
(ii) 

Theorem (Otago-S)

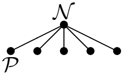
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

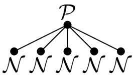
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i).

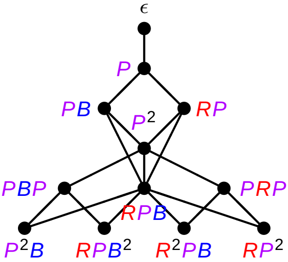
(i) 

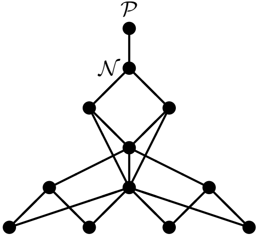
(ii) 

Theorem (Otago-S)

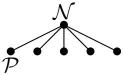
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

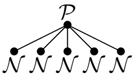
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$.

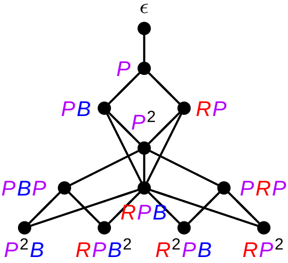
(i) 

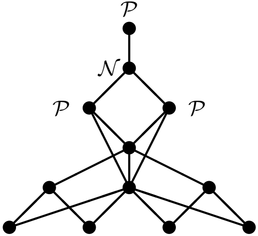
(ii) 

Theorem (Otago-S)

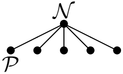
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

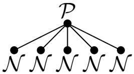
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$.

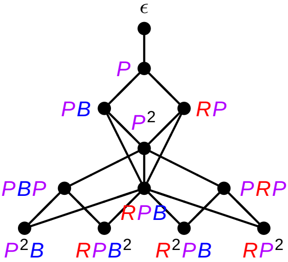
(i) 

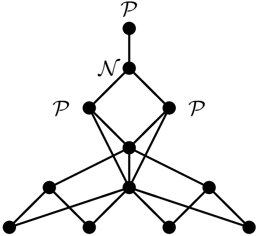
(ii) 

Theorem (Otago-S)

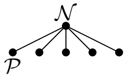
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

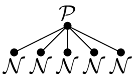
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$. So $\ell(P^2) = \ell(RPB) = \mathcal{N}$ by (ii).

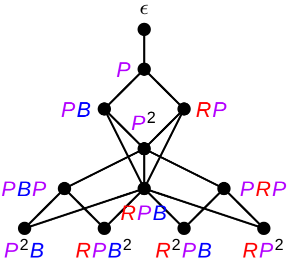
(i) 

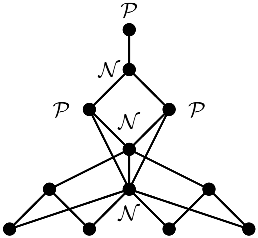
(ii) 

Theorem (Otago-S)

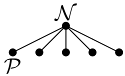
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

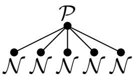
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$. So $\ell(P^2) = \ell(RPB) = \mathcal{N}$ by (ii).

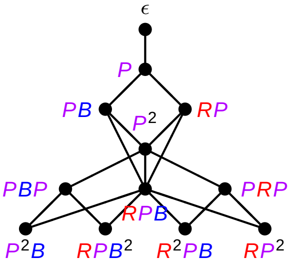
(i) 

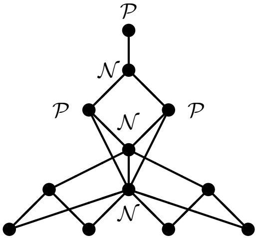
(ii) 

Theorem (Otago-S)

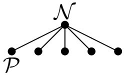
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

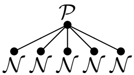
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$. So $\ell(P^2) = \ell(RPB) = \mathcal{N}$ by (ii). So $\ell(PBP) = \ell(PRP) = \mathcal{P}$ by (i) and symmetry.

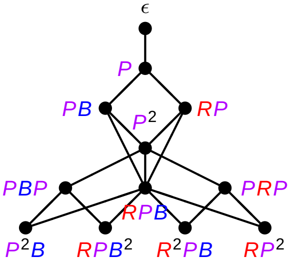
(i) 

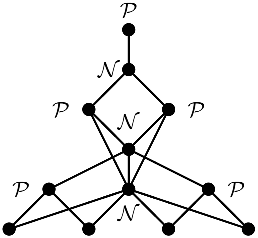
(ii) 

Theorem (Otago-S)

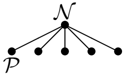
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

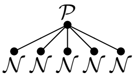
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$. So $\ell(P^2) = \ell(RPB) = \mathcal{N}$ by (ii). So $\ell(PBP) = \ell(PRP) = \mathcal{P}$ by (i) and symmetry.

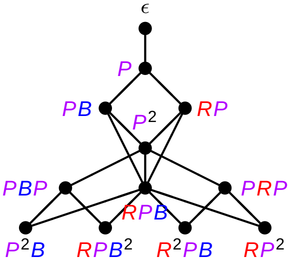
(i) 

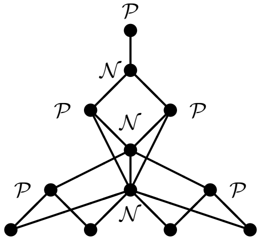
(ii) 

Theorem (Otago-S)

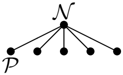
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

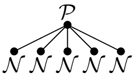
Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$. So $\ell(P^2) = \ell(RPB) = \mathcal{N}$ by (ii). So $\ell(PBP) = \ell(PRP) = \mathcal{P}$ by (i) and symmetry. So $\ell(P^2B) = \ell(RPB^2) = \ell(R^2PB) = \ell(RP^2) = \mathcal{N}$ by (ii).

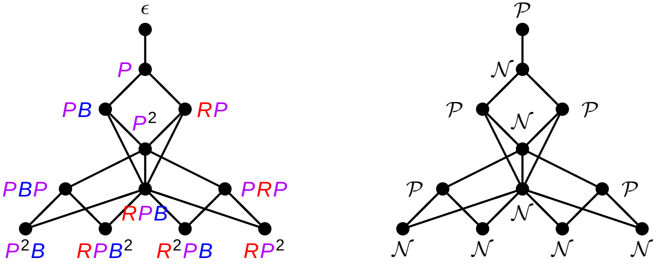
(i) 

(ii) 

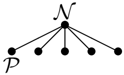
Theorem (Otago-S)

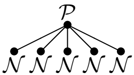
The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

Proof. Part of the top of $\overline{T}$ is



Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$. So $\ell(P^2) = \ell(RPB) = \mathcal{N}$ by (ii). So $\ell(PBP) = \ell(PRP) = \mathcal{P}$ by (i) and symmetry. So $\ell(P^2B) = \ell(RPB^2) = \ell(R^2PB) = \ell(RP^2) = \mathcal{N}$ by (ii).

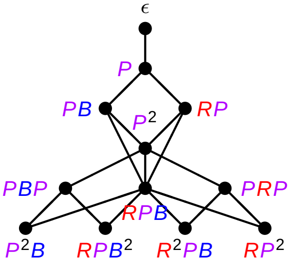
(i) 

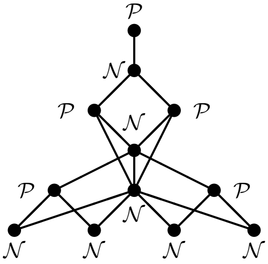
(ii) 

Theorem (Otago-S)

The winner of the game on $\mathbb{Q}$ where $m = n \geq 3$ is A.

Proof. Part of the top of $\overline{T}$ is





Suppose B wins, so $\ell(\epsilon) = \mathcal{P}$. So $\ell(P) = \mathcal{N}$ by (ii). So $\ell(PB) = \mathcal{P}$ or $\ell(RP) = \mathcal{P}$ by (i). But $m = n$ and PB, RP are symmetric so $\ell(PB) = \ell(RP) = \mathcal{P}$. So $\ell(P^2) = \ell(RPB) = \mathcal{N}$ by (ii). So $\ell(PBP) = \ell(PRP) = \mathcal{P}$ by (i) and symmetry. So $\ell(P^2B) = \ell(RPB^2) = \ell(R^2PB) = \ell(RP^2) = \mathcal{N}$ by (ii). This contradicts $\ell(RPB) = \mathcal{N}$ ■

Outline

(1) Who wins the game for general m, n on either $[mn+1]$ or $\mathbb{Q}$? In appears as if A wins except when m or n is small.

(1) Who wins the game for general m, n on either $[mn+1]$ or $\mathbb{Q}$? In appears as if A wins except when m or n is small.

(2) What can be said about playing on other partially ordered sets?

(1) Who wins the game for general m, n on either $[mn + 1]$ or $\mathbb{Q}$? In appears as if A wins except when m or n is small.

(2) What can be said about playing on other partially ordered sets?

Theorem (Otago-S)

If $N \geq mn + 1$ then the winner playing on the Boolean algebra B_N is B.

HAPPY
BIRTHDAY
ANDREAS!!