Duplicator Spoiler Games
DUP and SPOIL

$a < b$.
DUP and SPOIL

\[ a < b. \]
\[ L_a = \{1 < 2 < \cdots < a\} \]
DUP and SPOIL

\[ a < b. \]
\[ L_a = \{1 < 2 < \cdots < a\} \]
\[ L_b = \{1 < 2 < \cdots < b\} \]
a < b.

\[ L_a = \{1 < 2 < \cdots < a\} \]
\[ L_b = \{1 < 2 < \cdots < b\} \]

DUP is **cra-cra**! He thinks \( L_a \) and \( L_b \) are the same!
DUP and SPOIL

\[ a < b. \]
\[ L_a = \{1 < 2 < \cdots < a\} \]
\[ L_b = \{1 < 2 < \cdots < b\} \]
DUP is **cra-cra**! He thinks \( L_a \) and \( L_b \) are the same!

1. SPOIL wants to convince DUP that \( L_a \neq L_b \).
DUP and SPOIL

$a < b$.

$L_a = \{1 < 2 < \cdots < a\}$

$L_b = \{1 < 2 < \cdots < b\}$

DUP is cra-cra! He thinks $L_a$ and $L_b$ are the same!

1. SPOIL wants to convince DUP that $L_a \neq L_b$.
2. DUP wants to resist the attempt.
DUP and SPOIL

\[ a < b. \]

\[ L_a = \{ 1 < 2 < \cdots < a \} \]
\[ L_b = \{ 1 < 2 < \cdots < b \} \]

DUP is **cra-cra**! He thinks \( L_a \) and \( L_b \) are the same!

1. SPOIL wants to convince DUP that \( L_a \not\equiv L_b \).
2. DUP wants to resist the attempt.

We will call SPOIL S and DUP D to fit on slides.
**S** Tries to Convince **D** that \(L_a \neq L_b\)

---

**Parameter** \(k\) The number of rounds.
S Tries to Convince D that $L_a \neq L_b$

**Parameter**  $k$ The number of rounds.

1. **S** pick number in one orderings.
S Tries to Convince D that \( L_a \neq L_b \)

**Parameter** \( k \) The number of rounds.

1. **S** pick number in one orderings.
2. **D** pick number in OTHER ORDERING. D will try to pick a point that most **looks like** the other point.
S Tries to Convince D that $L_a \neq L_b$

Parameter $k$ The number of rounds.

1. S pick number in one orderings.
2. D pick number in OTHER ORDERING. D will try to pick a point that most looks like the other point.
3. Repeat for $k$ rounds.
S Tries to Convince D that $L_a \neq L_b$

**Parameter** $k$ The number of rounds.

1. **S** pick number in one orderings.
2. **D** pick number in OTHER ORDERING. D will try to pick a point that most looks like the other point.
3. Repeat for $k$ rounds.
4. This process creates a map between $k$ points of $L_a$ and $k$ points of $L_b$. 
S Tries to Convince D that $L_a \neq L_b$

Parameter $k$ The number of rounds.

1. **S** pick number in one orderings.
2. **D** pick number in OTHER ORDERING. D will try to pick a point that most looks like the other point.
3. Repeat for $k$ rounds.
4. This process creates a map between $k$ points of $L_a$ and $k$ points of $L_b$.
5. If this map is order preserving D wins, else S wins.
S Tries to Convince D that $L_a \not\equiv L_b$

Parameter $k$ The number of rounds.

1. S pick number in one orderings.
2. D pick number in OTHER ORDERING. D will try to pick a point that most looks like the other point.
3. Repeat for $k$ rounds.
4. This process creates a map between $k$ points of $L_a$ and $k$ points of $L_b$.
5. If this map is order preserving D wins, else S wins.

Bill plays a student $(L_3, L_4, 2), (L_3, L_4, 3)$
Want Optimal $k$

Since $L_a \neq L_b$, $S$ will win if $k$ is large enough.
Want Optimal $k$

Since $L_a \neq L_b$, $S$ will win if $k$ is large enough. We want to know the smallest $k$. 
Want Optimal $k$

Since $L_a \neq L_b$, $S$ will win if $k$ is large enough. We want to know the smallest $k$. We assume both players play perfectly.
Want Optimal $k$

Since $L_a \neq L_b$, $S$ will win if $k$ is large enough. We want to know the smallest $k$. We assume both players play perfectly. We want $k$ such that

1. $S$ beats $D$ in the $(L_a, L_b, k)$ game.
2. $D$ beats $S$ in the $(L_a, L_b, k - 1)$ game.
Want Optimal $k$

Since $L_a \neq L_b$, $S$ will win if $k$ is large enough.
We want to know the smallest $k$.
We assume both players play perfectly.
We want $k$ such that

1. $S$ beats $D$ in the $(L_a, L_b, k)$ game.
Want Optimal $k$

Since $L_a \neq L_b$, $S$ will win if $k$ is large enough.
We want to know the smallest $k$.
We assume both players play perfectly.
We want $k$ such that
1. $S$ beats $D$ in the $(L_a, L_b, k)$ game.
2. $D$ beats $S$ in the $(L_a, L_b, k - 1)$ game.
Try to determine:

1. Who wins (L₃, L₄, 2)? (2 moves).
2. Who wins (L₈, L₁₀, 3)? (3 moves).
3. GENERALLY: Who wins (Lₐ, Lₜ, k).
Students With Your Neighbors

Try to determine:

1. Who wins \((L_3, L_4, 2)\)? (2 moves).
Try to determine:

1. Who wins \((L_3, L_4, 2)\)? (2 moves).
2. Who wins \((L_8, L_{10}, 3)\)? (3 moves)
Try to determine:

1. Who wins \((L_3, L_4, 2)\)? (2 moves).
2. Who wins \((L_8, L_{10}, 3)\)? (3 moves)
3. GENERALLY: Who wins \((L_a, L_b, k)\).
Generalize

Can use any orderings $L$, $L'$
Generalize

Can use any orderings $L, L'$

1. $\mathbb{N}$ and $\mathbb{Q}$ are the usual orderings.
Can use any orderings $L, L'$

1. $\mathbb{N}$ and $\mathbb{Q}$ are the usual orderings.
2. $\mathbb{N}^*$ is the ordering $\cdots < 2 < 1 < 0$. 

Generalize
Can use any orderings $L$, $L'$

1. $\mathbb{N}$ and $\mathbb{Q}$ are the usual orderings.
2. $\mathbb{N}^*$ is the ordering $\cdots < 2 < 1 < 0$.
3. If $L$ is an ordering then $L^*$ is that ordering backwards.
Can use any orderings $L, L'$

1. $\mathbb{N}$ and $\mathbb{Q}$ are the usual orderings.
2. $\mathbb{N}^*$ is the ordering $\cdots < 2 < 1 < 0$.
3. If $L$ is an ordering then $L^*$ is that ordering backwards.

**Play a student $\mathbb{N}$ and $\mathbb{Z}$ with 1 move, 2 moves**
In all problems we want a $k$ such that condition holds.
Students with your Neighbors

In all problems we want a $k$ such that condition holds.

1. $D$ wins $(\mathbb{N}, \mathbb{Z}, k-1)$, $S$ wins $(\mathbb{N}, \mathbb{Z}, k)$. 
In all problems we want a $k$ such that condition holds.

1. D wins ($\mathbb{N}, \mathbb{Z}, k - 1$), S wins ($\mathbb{N}, \mathbb{Z}, k$).
2. D wins ($\mathbb{N}, \mathbb{Q}, k - 1$), S wins ($\mathbb{N}, \mathbb{Q}, k$).
Students with your Neighbors

In all problems we want a $k$ such that condition holds.

1. D wins ($\mathbb{N}, \mathbb{Z}, k - 1$), S wins ($\mathbb{N}, \mathbb{Z}, k$).
2. D wins ($\mathbb{N}, \mathbb{Q}, k - 1$), S wins ($\mathbb{N}, \mathbb{Q}, k$).
3. D wins ($\mathbb{Z}, \mathbb{Q}, k - 1$), S wins ($\mathbb{Z}, \mathbb{Q}, k$).
Students with your Neighbors

In all problems we want a $k$ such that condition holds.

1. D wins $(\mathbb{N}, \mathbb{Z}, k - 1)$, S wins $(\mathbb{N}, \mathbb{Z}, k)$.
2. D wins $(\mathbb{N}, \mathbb{Q}, k - 1)$, S wins $(\mathbb{N}, \mathbb{Q}, k)$.
3. D wins $(\mathbb{Z}, \mathbb{Q}, k - 1)$, S wins $(\mathbb{Z}, \mathbb{Q}, k)$.
4. D wins $(L_{10}, \mathbb{N} + \mathbb{N}^*, k - 1)$, S wins $(L_{10}, \mathbb{N} + \mathbb{N}^*, k)$.
Students with your Neighbors

In all problems we want a \( k \) such that condition holds.

1. D wins \((\mathbb{N}, \mathbb{Z}, k - 1)\), S wins \((\mathbb{N}, \mathbb{Z}, k)\).
2. D wins \((\mathbb{N}, \mathbb{Q}, k - 1)\), S wins \((\mathbb{N}, \mathbb{Q}, k)\).
3. D wins \((\mathbb{Z}, \mathbb{Q}, k - 1)\), S wins \((\mathbb{Z}, \mathbb{Q}, k)\).
4. D wins \((L_{10}, \mathbb{N} + \mathbb{N}^*, k - 1)\), S wins \((L_{10}, \mathbb{N} + \mathbb{N}^*, k)\).
5. D wins \((\mathbb{N} + \mathbb{Z}, \mathbb{N}, k - 1)\), S wins \((\mathbb{N} + \mathbb{Z}, \mathbb{N}, k)\).
A Notion of $L, L'$ being Similar

Let $L$ and $L'$ be two linear orderings.
A Notion of $L, L'$ being Similar

Let $L$ and $L'$ be two linear orderings.

**Def** If $D$ wins the $k$-round DS-game on $L, L'$ then $L, L'$ are $k$-game equivalent (denoted $L \equiv^G_k L'$).
What is Truth?

All sentences use the usual logic symbols and $\prec$. 
What is Truth?

All sentences use the usual logic symbols and $\prec$.

**Def** If $L$ is a linear ordering and $\phi$ is a sentence then $L \models \phi$ means that $\phi$ is true in $L$. 
What is Truth?

All sentences use the usual logic symbols and <.

**Def** If $L$ is a linear ordering and $\phi$ is a sentence then $L \models \phi$ means that $\phi$ is true in $L$.

**example** Let $\phi = (\forall x)(\forall y)(\exists z)[x < y \implies x < z < y]$
All sentences use the usual logic symbols and <.

**Def** If $L$ is a linear ordering and $\phi$ is a sentence then $L \models \phi$ means that $\phi$ is true in $L$.

**Example** Let $\phi = (\forall x)(\forall y)(\exists z)[x < y \implies x < z < y]$

1. $\mathbb{Q} \models \phi$
What is Truth?

All sentences use the usual logic symbols and <.

**Def** If $L$ is a linear ordering and $\phi$ is a sentence then $L \models \phi$ means that $\phi$ is true in $L$.

**Example** Let $\phi = (\forall x)(\forall y)(\exists z)[x < y \implies x < z < y]$

1. $\mathbb{Q} \models \phi$
2. $\mathbb{N} \models \neg \phi$
Quantifier Depth Formally

If \( \phi(\vec{x}) \) has 0 quantifiers then \( qd(\phi(\vec{x})) = 0 \).
Quantifier Depth Formally

If $\phi(\vec{x})$ has 0 quantifiers then $qd(\phi(\vec{x})) = 0$.
If $\alpha \in \{\land, \lor, \rightarrow\}$ then
Quantifier Depth Formally

If \( \phi(\vec{x}) \) has 0 quantifiers then \( qd(\phi(\vec{x})) = 0. \)

If \( \alpha \in \{\land, \lor, \rightarrow\} \) then

\[
qd(\phi_1(\vec{x}) \alpha \phi_2(\vec{x})) = \max\{qd(\phi_1(\vec{x}), qd(\phi_2(\vec{x}))\).
\]

If \( Q \in \{\exists, \forall\} \) then

\[
qd((Qx_1)[\phi(x_1, \ldots, x_n)]) = qd(\phi_1(x_1, \ldots, x_n)) + 1.
\]
Quantifier Depth Formally

If $\phi(\vec{x})$ has 0 quantifiers then $qd(\phi(\vec{x})) = 0$.

If $\alpha \in \{\land, \lor, \to\}$ then

$$qd(\phi_1(\vec{x}) \alpha \phi_2(\vec{x})) = \max\{qd(\phi_1(\vec{x})), qd(\phi_2(\vec{x}))\}.$$ 

$$qd(\neg \phi(\vec{x})) = qd(\phi(\vec{x})).$$
Quantifier Depth Formally

If $\phi(\vec{x})$ has 0 quantifiers then $qd(\phi(\vec{x})) = 0$.

If $\alpha \in \{\land, \lor, \rightarrow\}$ then

$$qd(\phi_1(\vec{x}) \alpha \phi_2(\vec{x})) = \max\{qd(\phi_1(\vec{x})), qd(\phi_2(\vec{x}))\}.$$ 

$$qd(\neg \phi(\vec{x})) = qd(\phi(\vec{x})).$$

If $Q \in \{\exists, \forall\}$ then

$$qd((Qx_1)[\phi(x_1, \ldots, x_n)] = qd(\phi_1(x_1, \ldots, x_n)) + 1.$$
Example of Quantifier Depth

\[(\forall x)(\forall z)[x < z \rightarrow (\exists y)[x < y < z]]\]
Example of Quantifier Depth

$$(\forall x)(\forall z)[x < z \rightarrow (\exists y)[x < y < z]]$$

Let's take it apart
Example of Quantifier Depth

$$(\forall x)(\forall z)[x < z \rightarrow (\exists y)[x < y < z]]$$

Lets take it apart

$qd((\exists y)[x < y < z]) = 1 + 0 = 1.$
Example of Quantifier Depth

\((\forall x)(\forall z)[x < z \rightarrow (\exists y)[x < y < z]]\)

Lets take it apart

qd((\exists y)[x < y < z])) = 1 + 0 = 1.

qd(x < z \rightarrow (\exists y)[x < y < z]) = \max\{0, 1\} = 1.
Example of Quantifier Depth

$$(\forall x)(\forall z)[x < z \rightarrow (\exists y)[x < y < z]]$$

Let's take it apart

$qd((\exists y)[x < y < z]) = 1 + 0 = 1.$

$qd(x < z \rightarrow (\exists y)[x < y < z]) = \max\{0, 1\} = 1.$

$qd((\forall x)(\forall z)[x < z \rightarrow (\exists y)[x < y < z]]) = 2 + 1 = 3$
Another Notion of $L, L'$ Similar

Let $L$ and $L'$ be two linear orderings.
Another Notion of $L, L'$ Similar

Let $L$ and $L'$ be two linear orderings.

**Def** $L$ and $L'$ are $k$-truth-equiv ($L \equiv^T_k L'$)

$$(\forall \phi, \text{qd}(\phi) \leq k)[L \models \phi \iff L' \models \phi].$$
The Big Theorem

**Thm** Let $L, L'$ be any linear ordering and let $k \in \mathbb{N}$. 
**Thm** Let $L, L'$ be any linear ordering and let $k \in \mathbb{N}$. The following are equivalent.
The Big Theorem

**Thm** Let $L, L'$ be any linear ordering and let $k \in \mathbb{N}$. The following are equivalent.

1. $L \equiv_k L'$
Theorem Let $L, L'$ be any linear ordering and let $k \in \mathbb{N}$. The following are equivalent.

1. $L \equiv_k^T L'$
2. $L \equiv_k^G L'$
Applications

1. Density cannot be expressed with $qd^2$.
   (Proof: $Z \equiv G^2 Q \Rightarrow Z \equiv T^2 Q$).

2. Well foundedness cannot be expressed in 1st order at all!
   (Proof: $(\forall n)[N^+ Z \equiv G^2 n N]$).

3. Upshot: Questions about expressability become questions about games.

4. Complexity: As Computer Scientists we think of complexity in terms of time or space (e.g., sorting $n$ elements can be done in roughly $n \log n$ comparisons). But how do you measure complexity for concepts where time and space do not apply? One measure is quantifier depth. These games help us prove LOWER BOUNDS on quantifier depth!
Applications

1. Density *cannot* be expressed with $qd \ 2$.  
   (Proof: $\mathbb{Z} \equiv^G_2 \mathbb{Q}$ so $\mathbb{Z} \equiv^T_2 \mathbb{Q}$).

2. Well foundedness cannot be expressed in 1st order at all!  
   (Proof: $(\forall n)[\mathbb{N} + \mathbb{Z} \equiv G n \mathbb{N}]$).

WILL DO ON WHITE BOARD.

3. Upshot: Questions about expressability become questions about games.

4. Complexity: As Computer Scientists we think of complexity in terms of time or space (e.g., sorting $n$ elements can be done in roughly $n \log n$ comparisons). But how do you measure complexity for concepts where time and space do not apply? One measure is quantifier depth. These games help us prove LOWER BOUNDS on quantifier depth!
Applications

1. Density \textit{cannot} be expressed with \textit{qd} 2.
   (Proof: $\mathbb{Z} \equiv^G \mathbb{Q}$ so $\mathbb{Z} \equiv^T \mathbb{Q}$).

2. Well foundedness cannot be expressed in 1st order at all!
   (Proof: $(\forall n)[\mathbb{N} + \mathbb{Z} \equiv^G \mathbb{N}]$).
   WILL DO ON WHITE BOARD.

3. Upshot: Questions about expressability become questions about games.

4. Complexity: As Computer Scientists we think of complexity in terms of time or space (e.g., sorting $n$ elements can be done in roughly $n \log n$ comparisons). But how do you measure complexity for concepts where time and space do not apply? One measure is quantifier depth. These games help us prove LOWER BOUNDS on quantifier depth!
Applications

1. Density cannot be expressed with qd 2.
   (Proof: $\mathbb{Z} \equiv_2^G \mathbb{Q}$ so $\mathbb{Z} \equiv_2^T \mathbb{Q}$).

2. Well foundedness cannot be expressed in 1st order at all!
   (Proof: $(\forall n)[\mathbb{N} + \mathbb{Z} \equiv_n^G \mathbb{N}]$).
   WILL DO ON WHITE BOARD.

3. Upshot: Questions about expressability become questions about games.
Applications

1. Density cannot be expressed with $qd\ 2$. (Proof: $\mathbb{Z} \equiv^G_2 \mathbb{Q}$ so $\mathbb{Z} \equiv^T_2 \mathbb{Q}$).

2. Well foundedness cannot be expressed in 1st order at all! (Proof: $(\forall n)[\mathbb{N} + \mathbb{Z} \equiv^G_n \mathbb{N}]$). WILL DO ON WHITE BOARD.

3. Upshot: Questions about expressability become questions about games.

4. Complexity: As Computer Scientists we think of complexity in terms of time or space (e.g., sorting $n$ elements can be done in roughly $n \log n$ comparisons). But how do you measure complexity for concepts where time and space do not apply? One measure is quantifier depth. These games help us prove LOWER BOUNDS on quantifier depth!
Proving DUP Wins Rigorously
Notation

The game where the orders are $L$ and $L'$, and its for $n$ moves, will be denoted

$$(L, L'; n)$$
\( L_a \) and \( L_b \)

**Thm** For all \( n \), if \( a, b \geq 2^n \) then DUP wins \((L_a, L_b; n)\).
$L_a$ and $L_b$

**Thm** For all $n$, if $a, b \geq 2^n$ then DUP wins $(L_a, L_b; n)$.

**IB** $n = 1$. DUP clearly wins $(L_a, L_b; 1)$.
**La and Lb**

**Thm** For all \( n \), if \( a, b \geq 2^n \) then DUP wins \((L_a, L_b; n)\).

**IB** \( n = 1 \). DUP clearly wins \((L_a, L_b; 1)\).

**IH** For all \( a, b \geq 2^{n-1} \), DUP wins \((L_a, L_b; n - 1)\).
**Thm** For all $n$, if $a, b \geq 2^n$ then DUP wins $(L_a, L_b; n)$.

**IB** $n = 1$. DUP clearly wins $(L_a, L_b; 1)$.

**IH** For all $a, b \geq 2^{n-1}$, DUP wins $(L_a, L_b; n - 1)$.

**IS** We do 1 case: SP makes move $x \leq 2^{n-1}$ in $L_a$.

DUP respond with $x$ in $L_b$. DUP views game as 2 GAMES:
**La and Lb**

**Thm** For all $n$, if $a, b \geq 2^n$ then DUP wins $(L_a, L_b; n)$.

**IB** $n = 1$. DUP clearly wins $(L_a, L_b; 1)$.

**IH** For all $a, b \geq 2^{n-1}$, DUP wins $(L_a, L_b; n - 1)$.

**IS** We do 1 case: SP makes move $x \leq 2^{n-1}$ in $L_a$. DUP respond with $x$ in $L_b$. DUP views game as 2 GAMES:

**Key** The game is now 2 games.
$L_a$ and $L_b$

**Thm** For all $n$, if $a, b \geq 2^n$ then DUP wins $(L_a, L_b; n)$.

**IB** $n = 1$. DUP clearly wins $(L_a, L_b; 1)$.

**IH** For all $a, b \geq 2^{n-1}$, DUP wins $(L_a, L_b; n - 1)$.

**IS** We do 1 case: SP makes move $x \leq 2^{n-1}$ in $L_a$.

DUP respond with $x$ in $L_b$. DUP views game as 2 GAMES:

**Key** The game is now 2 games.

- $< x$ in both orders: $(L_{x-1}, L_{x-1}; n - 1)$. SP will never play here.
**Thm** For all $n$, if $a, b \geq 2^n$ then DUP wins $(L_a, L_b; n)$.

**IB** $n = 1$. DUP clearly wins $(L_a, L_b; 1)$.

**IH** For all $a, b \geq 2^{n-1}$, DUP wins $(L_a, L_b; n - 1)$.

**IS** We do 1 case: SP makes move $x \leq 2^{n-1}$ in $L_a$.

DUP respond with $x$ in $L_b$. DUP views game as 2 GAMES:

**Key** The game is now 2 games.

- $< x$ in both orders: $(L_{x-1}, L_{x-1}; n - 1)$. SP will never play here.

- $> x$ in both orders: $(L_{a-x}, L_{b-x}; n - 1)$.
**$L_a$ and $L_b$**

**Thm** For all $n$, if $a, b \geq 2^n$ then DUP wins $(L_a, L_b; n)$.

**IB** $n = 1$. DUP clearly wins $(L_a, L_b; 1)$.

**IH** For all $a, b \geq 2^{n-1}$, DUP wins $(L_a, L_b; n - 1)$.

**IS** We do 1 case: SP makes move $x \leq 2^{n-1}$ in $L_a$. DUP respond with $x$ in $L_b$. DUP views game as 2 GAMES:

**Key** The game is now 2 games.

- $< x$ in both orders: $(L_{x-1}, L_{x-1}; n - 1)$. SP will never play here.

- $> x$ in both orders: $(L_{a-x}, L_{b-x}; n - 1)$.
  Since $x \leq 2^{n-1}$ and $a, b \geq 2^n$, $a - x - 1 \geq 2^{n-1}$ and $b - x - 1 \geq 2^{n-1}$. 

**La and Lb**

**Thm** For all $n$, if $a, b \geq 2^n$ then DUP wins $(L_a, L_b; n)$.

**IB** $n = 1$. DUP clearly wins $(L_a, L_b; 1)$.

**IH** For all $a, b \geq 2^{n-1}$, DUP wins $(L_a, L_b; n - 1)$.

**IS** We do 1 case: SP makes move $x \leq 2^{n-1}$ in $L_a$.

DUP respond with $x$ in $L_b$. DUP views game as 2 GAMES:

**Key** The game is now 2 games.

- $< x$ in both orders: $(L_{x-1}, L_{x-1}; n - 1)$. SP will never play here.

- $> x$ in both orders: $(L_{a-x}, L_{b-x}; n - 1)$.

Since $x \leq 2^{n-1}$ and $a, b \geq 2^n$, $a - x - 1 \geq 2^{n-1}$ and $b - x - 1 \geq 2^{n-1}$.

By IH DUP wins $(L_{a-x}, L_{b-x}; n - 1)$. 
General Principle

1. After the 1st move \( x \) in \( L \) and the counter-move \( x' \) in \( L' \), the game is now two boards,
   1.1 \( L < x \) and \( L' < x' \).
   1.2 \( L > x \) and \( L' > x' \).

2. We might use induction on those smaller boards.

3. Might not need induction on the smaller boards if they are orderings we already proved things about.
General Principle

1. After the 1st move $x$ in $L$ and the counter-move $x'$ in $L'$, the game is now two boards,
General Principle

1. After the 1st move $x$ in $L$ and the counter-move $x'$ in $L'$, the game is now two boards,
   1.1 $L < x$ and $L' < x'$.
General Principle

1. After the 1st move $x$ in $L$ and the counter-move $x'$ in $L'$, the game is now two boards,
   1.1 $L < x$ and $L' < x'$.
   1.2 $L > x$ and $L' > x'$.
General Principle

1. After the 1st move \( x \) in in \( L \) and the counter-move \( x' \) in \( L' \), the game is now two boards,
   
   1.1 \( L <^x \) and \( L' <^{x'} \).
   
   1.2 \( L >^x \) and \( L' >^{x'} \).

2. We might use induction on those smaller boards.
General Principle

1. After the 1st move $x$ in in $L$ and the counter-move $x'$ in $L'$, the game is now two boards,
   1.1 $L < x$ and $L' < x'$.
   1.2 $L > x$ and $L' > x'$.

2. We might use induction on those smaller boards.

3. Might not need induction on the smaller boards if they are orderings we already proved things about.
\( N + N^* \) and \( L_a \)

**Thm** For all \( n \), if \( a \geq 2^n \), DUP wins \( (N + N^*, L_a; n) \).
$\mathbb{N} + \mathbb{N}^*$ and $L_a$

**Thm** For all $n$, if $a \geq 2^n$, DUP wins $(\mathbb{N} + \mathbb{N}^*, L_a; n)$. Might make this a HW.
\textbf{Thm} For all \( n \), DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n)\).
**Thm** For all $n$, DUP wins $(\mathbb{N}, \mathbb{N} + \mathbb{Z}; n)$.

**IB** $n = 1$. DUP clearly wins $(\mathbb{N}, \mathbb{N} + \mathbb{Z}; 1)$. 

**N and N + Z**
\textbf{Thm} For all \( n \), DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n)\).

\textbf{IB} \( n = 1 \). DUP clearly wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; 1)\).

\textbf{IH} DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)\).
For all \( n \), DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n)\).

**IB** \( n = 1 \). DUP clearly wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; 1)\).

**IH** DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)\).

1) SP plays \( x \) in either \( \mathbb{N} \) or \( \mathbb{N} \)-part of \( \mathbb{N} + \mathbb{Z} \) then DUP counters with the same \( x \) in the other part. The 2 games are
Thm For all \( n \), DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n)\).

IB \( n = 1 \). DUP clearly wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; 1)\).

IH DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)\).

1) SP plays \( x \) in either \( \mathbb{N} \) or \( \mathbb{N} \)-part of \( \mathbb{N} + \mathbb{Z} \) then DUP counters with the same \( x \) in the other part. The 2 games are \((L_x, L_x; n - 1)\) and \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)\).

SP won't play on 1st board.

The 2nd board \( DUP \) wins by IH.
Theorem For all $n$, DUP wins $(\mathbb{N}, \mathbb{N} + \mathbb{Z}; n)$.

Base Case $n = 1$. DUP clearly wins $(\mathbb{N}, \mathbb{N} + \mathbb{Z}; 1)$.

Inductive Hypothesis DUP wins $(\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)$.

1) SP plays $x$ in either $\mathbb{N}$ or $\mathbb{N}$-part of $\mathbb{N} + \mathbb{Z}$ then DUP counters with the same $x$ in the other part. The 2 games are $(L_x, L_x; n - 1)$ and $(\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)$.

SP won’t play on 1st board.

The 2nd board $DUP$ wins by IH.

2) SP plays $x$ in $\mathbb{Z}$ part of $\mathbb{N} + \mathbb{Z}$ then DUP plays $2^n$ in $\mathbb{N}$. The 2 games are
**Thm** For all \( n \), DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n)\).

**IB** \( n = 1 \). DUP clearly wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; 1)\).

**IH** DUP wins \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)\).

1) SP plays \( x \) in either \( \mathbb{N} \) or \( \mathbb{N} \)-part of \( \mathbb{N} + \mathbb{Z} \) then DUP counters with the same \( x \) in the other part. The 2 games are \((L_x, L_x; n - 1)\) and \((\mathbb{N}, \mathbb{N} + \mathbb{Z}; n - 1)\).

SP won’t play on 1st board.

The 2nd board **DUP** wins by IH.

2) SP plays \( x \) in \( \mathbb{Z} \) part of \( \mathbb{N} + \mathbb{Z} \) then DUP plays \( 2^n \) in \( \mathbb{N} \). The 2 games are \((\mathbb{N} + \mathbb{N}^*, L_{2^n}; n - 1)\) and \((\mathbb{N}, \mathbb{N}; n - 1)\).

SP won’t play on 2nd board. DUP wins 1st board by prior thm.