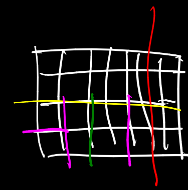
Groups

U - 3/4 people - 1 paper

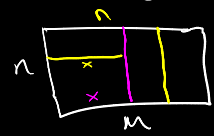
M - 2 people - 2 papers

# Strategy

- Symmetry
- Ex. Chop.

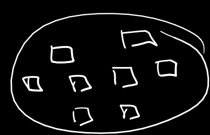


$m \times n$ grid $\rightarrow n \times n$ - - - - $\rightarrow 1 \times 1$

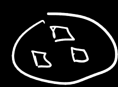



win by always cutting it to a square

Ex. PUB



pick up 1 or 2



- 2nd player always wins

If 3 divides $n$, then the 2nd player has a winning strategy.

Otherwise, the first player has a winning strategy.

## Strategy Stealing

- prove existence w/o knowing the strategy

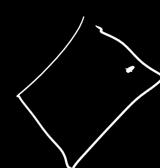
Thm. Hex is a first player win. → First player has a winning strategy

Pf. Assume 2nd player has a winning strategy.

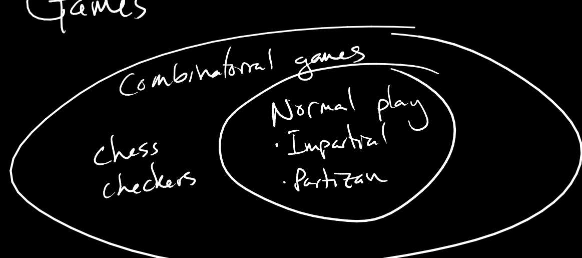
- 1st player places first piece randomly.
- Then pretends to be 2nd player and use the 2nd player winning strategy.
- Extra piece only benefits
- If strategy says to move in a spot you've already played. place another random piece
- Now you have the 2nd player winning strategy.

  # contradiction since you're the 1st player.

→ 1st player has a winning strategy.



# Normal Play Games




Impartial - same moves available to both player

Partizan - different moves available to each player

# Positions and Types

$G = \{L \mid R\}$

partizans left right

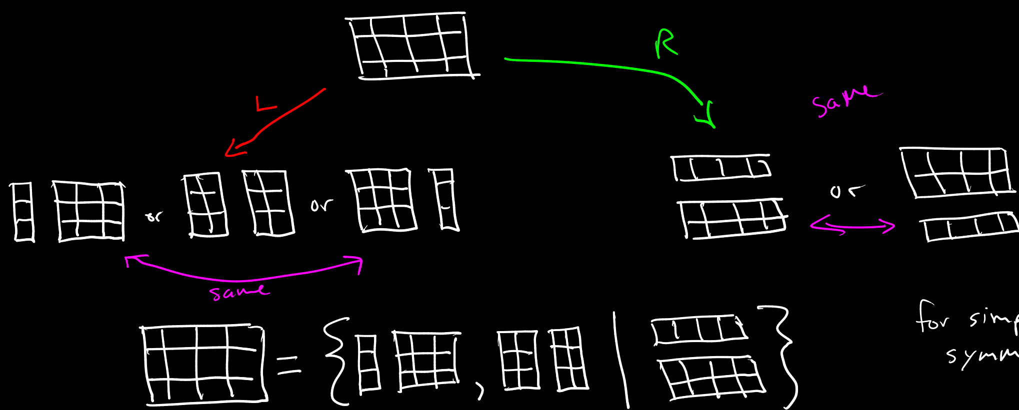
Impartial

$G = \{\_\_\}$

$G = \{G^L \mid G^R\}$

Cut-cake

- L vertical
- R Horizontal




for simplicity, we ignore symmetric equivalent moves.

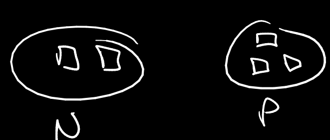
- Zermelo's Thm.

  Normal play games can not end in a draw.

Corollary: Every position in a normal-play game is one of the following types.

| Type | Description |
|---|---|
| L | Left (Louise) has a winning strategy whoever goes first |
| R | Right (Richard) has a winning strategy whoever goes first. |
| N | The Next player to play has a winning strategy. |
| P | The second or Previous player has a winning strategy |

Ex. PUB




cut cake


Determining Type

using recursive analysis

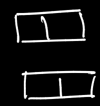
Proposition. If $\gamma = \{\alpha_1, \ldots, \alpha_m \mid \beta_1, \ldots, \beta_n\}$, the type of $\gamma$ is given by

| | some $\beta_j$ is type R or P | all of $\beta_1, \ldots, \beta_n$ are types L or N |
|---|---|---|
| Some $\alpha_i$ is type L or P | N | L |
| all of $\alpha_1, \ldots, \alpha_m$ are types R or N | R | P |

If you're N, you want a type P position

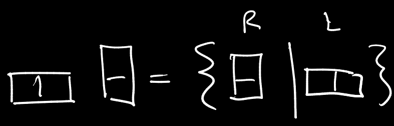
# Cut-cake Example

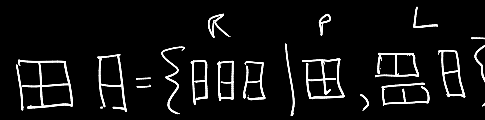
 has no moves - type P

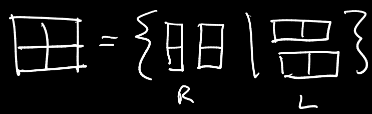

type L

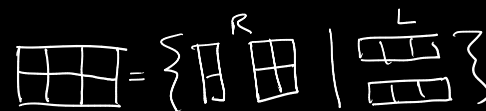

type R

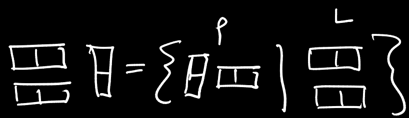
- L vertical
- R Horizontal

 $\Rightarrow$ type P

 $\Rightarrow$ type R

 $\Rightarrow$ type P

 $\Rightarrow$ type P

 $\Rightarrow$ type L