CGT class notes 2024

this is a brief lecture summary

I suggest that you also take your own notes

lecture 1

- **syllabus**
- **linear clobber**
- **who wins** `xoxoxoxoxoxo` **?**
- **cgt, egt**

clobber

L player Left (also bLack, x)
R player Right (also whiTe, o)

N-psn (first player win)
P-psn (first player loss)
L-psn (L wins as 1st and 2nd player)
R-psn (R wins as 1st and 2nd player)

e.g. clobber

ox   in N

oxox ?

assume w.l.o.g. L plays first

tree of all possible continuations of the game ?

say x plays first

```
             oxox
          /    |    \
        x_ox  o_xx  oxx_
          |           |
        x__o        _ox_
                      |
                    _x__
```

lecture 2

Zermelo's theorem

G: 2-player alt-turn finite (so no loops) win/loss/draw game

one of these holds:

- **ptm wins (means: exists winning strategy for (ptm, G))**
- **optm wins**
- **ptm draws and optm draws**
  - **exists at-least-draw strat for (ptm,G), and**
  - **exists at-least-draw strat for (optm,G)**

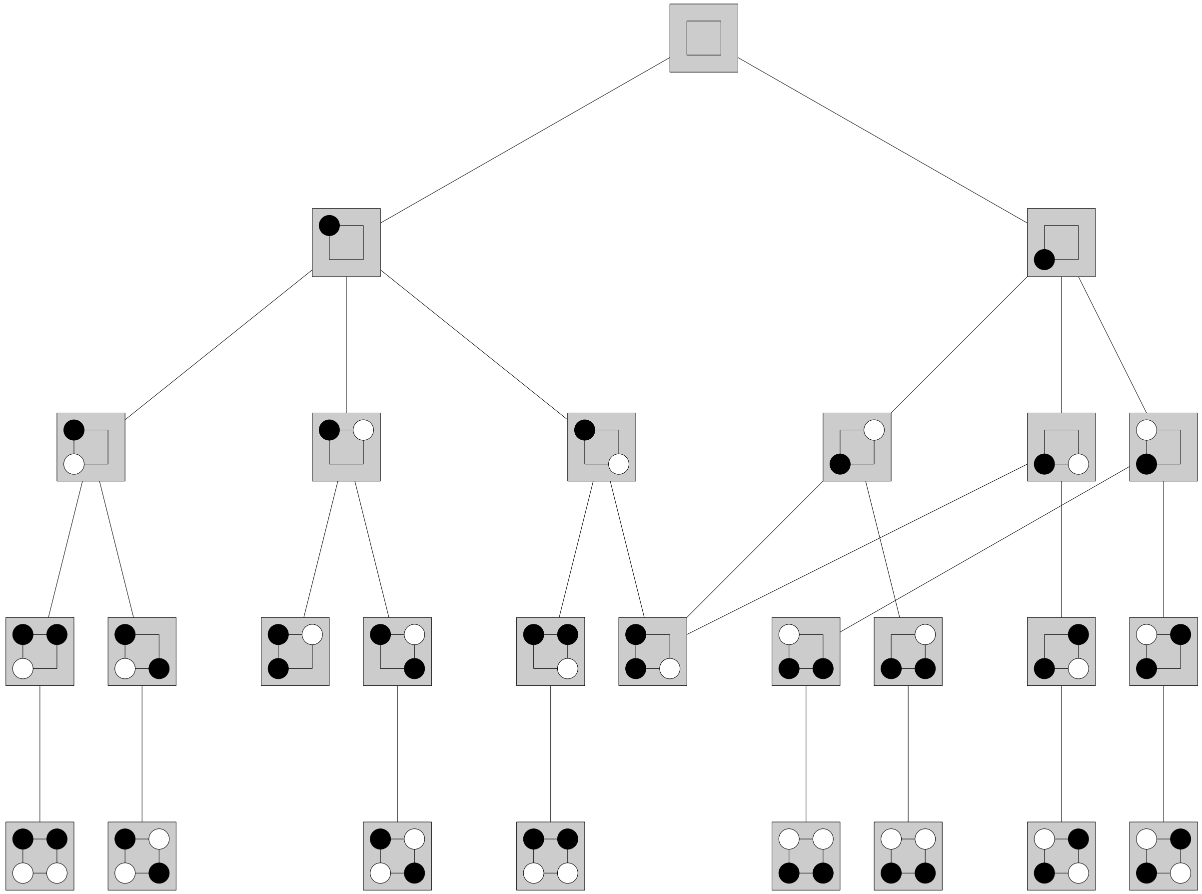
a simple game:

2p alt-turn

B wins with vertical column

W wins with horizontal row

next page: game dag (directed acyclic graph)



sketch of proof of Z's theorem

now that we have seen examples on the diagram, here is an outline of how to proof Z's thm...

$*$ inductively apply Z's thm for each subgame whose game dag has its root as a child of the original dag's root

$*$ this gives us a value of -1, 0, or 1 for each child of the root

sketch of proof of Z's theorem (continued)

how do we prove Z's thm at the root?

case 1: min values of children $= -1$

then value of root is 1 :)

case 2: min values of children = 0

then value of root is 0

root-ptm's drawing strategy?

- play to such a child

root-optm's drawing strategy ?

- exercise

case 3: min values of children = 1

then value of root is -1

root-optm's winning strategy?

easy: for each root-ptm move, we end up at a child

with value 1 and it's root-optm's move

mwah haaaahhhhaaaaaa

back to linear clobber

who wins `ox12` **?** `(oxoxoxoxoxox)`

game `x { | }` **call this game 0**

game `ox { 0 | 0 }` **call this game** $*$

game `oxx { 0 | * }` **call this game up**

game `oox { * | 0 }` **call this game down**

game `oxox { 0, *, up | 0, *, down }`