

Konečné hry s úplnou informací

hra dvou hráčů

konečná...skončí po konečném počtu kroků

s úplnou informací...oba hráči mají

všechny informace o stavu hry

Příklad: odebírání zápalek, piškvorky na konečné ploše, dáma, šachy, go...

Vyhrávající a prohrávající pozice

(kdyby neexistovala remíza)

- koncová pozice, ve které hráč na tahu prohrál,
je **prohrávající**
 - koncová pozice, ve které hráč na tahu vyhrál,
je **vyhrávající**
-

- pozice, ze které **lze** hráče dostat do prohrávající pozice,
je **vyhrávající**
- pozice, ze které **nelze** hráče dostat do prohrávající pozice
(tj. všechny tahy vedou do vyhrávající pozice soupeře),
je **prohrávající**

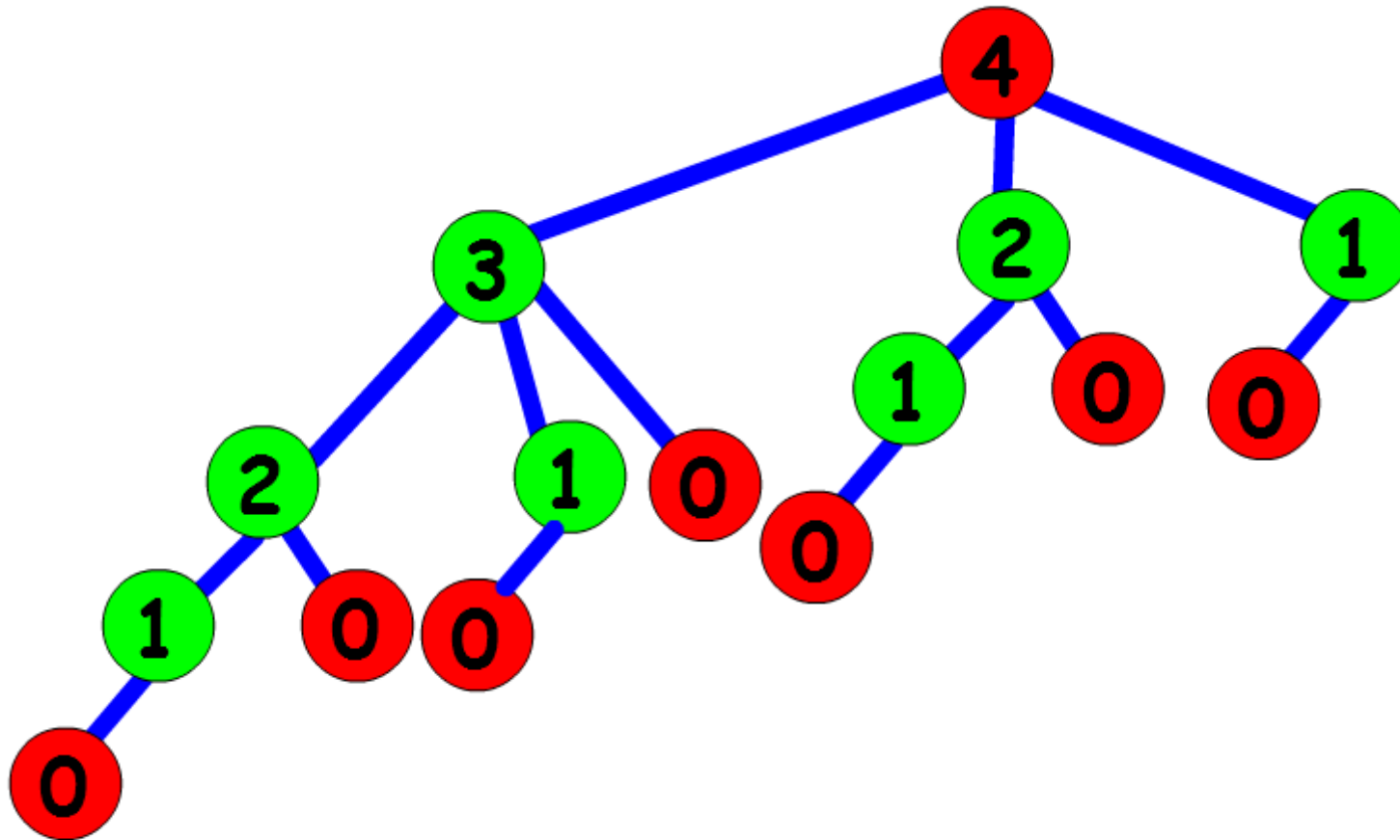
=> (kdyby neexistovala remíza)

každá dostupná pozice je buď vyhrávající nebo prohrávající

Strom hry

(nemusí být strom)

= Graf, vrcholy - stavy hry, hrany - možné tahy



Pokud existuje remíza, jsou kromě prohrávajících a vyhrávajících pozic ještě **neprohrávající** pozice.

Věta:

U konečné hry s úplnou informací alespoň pro jednoho hráče existuje neprohrávající strategie.

příklad: zápalky 1..3

příklad: zápalky z více hromádek

Hry s ohodnocením

= Každá cílová pozice je ohodnocena číslem
(namísto vyhrál-prohrál).

Jeden hráč se snaží dosáhnout maximálního výsledku,
druhý minimálního.

Hra s nulovým součtem

= Hra, ve které zisk jednoho hráče
je roven ztrátě druhého hráče.

Algoritmus MINIMAX

Strom hry se ohodnocuje od listů - koncových pozic,
(tam už víme, jak hra dopadla a ohodnocení známe).

Hodnota vrcholu-stavu se spočte jako
maximum nebo minimum hodnot synů,
podle toho,

zda je na tahu

maximalizující nebo minimalizující hráč.

příklad: řada čísel - součet dolů vs. součet nahoru

příklad: tabulka čísel - součet dolů vs. součet nahoru

Negamax

= Alternativa MINIMAXu,
namísto střídání MIN a MAX se počítá $-MAX(-...)$.
Snazší naprogramování.

$H := \max(\min(\max(\min(\dots))))$

vs.

$H := \max(-\max(-\max(-\max(-\dots))))$

alfa-beta prořezávání

Představme si, že

V uzlu **U** vybíráme maximum

a už umíme získat hodnotu **ALFA**

V jeho synu **V** vybíráme minimum...

a našli jsme hodnotu **x** menší nebo rovnu **ALFA**

=> nemá smysl zkoumat další tahy z uzlu **V** !

Protože:

- hodnota uzlu **V** bude určitě $\leq x$ (a $x \leq \text{ALFA}$)
...a dalším hledáním ji můžeme jedině zmenšit
- hodnota uzlu **U** bude určitě $\geq \text{ALFA}$
...takže ji hodnota uzlu **V** už nezvýší

Analogicky hodnota **BETA** a výběr minimum-maximum.

"Skutečné" hry

- omezená hloubka a statická ohodnocovací funkce
- hloubka ne všude stejná
- alfa-beta prořezávání
- okénko
- procházení do rostoucí hloubky
- příklad: `BP70\EXAMPLES\CHESS`

Heuristika

= rada, něco, co „obvykle dává dobré výsledky“

V užším smyslu (nic nezkazí)

pořadí prozkumávání tahů...

V širším smyslu (náhrada úplného řešení).

příklad s loupežníky...

