

## **BI-PYT, Ivan Romanenko ( romaniv1 ), semestralka.**

Cílem semestralky je napsat tahovou capture-strategy hru s AI.

Dál je popis defaultní hry. Pak je popis toho, co bude umožňovat kód ( virtualizace pro jednoduchou změnu různých částí hry ), jaké algoritmické řešení budou použita apod.

//-----//

### **Defaultní hra:**

Hra se hraje na celočíselné 2D mřížce 6x8. Celkem hraje 2 hráče ( člověk vs. AI ), které postupně dělají tahy.

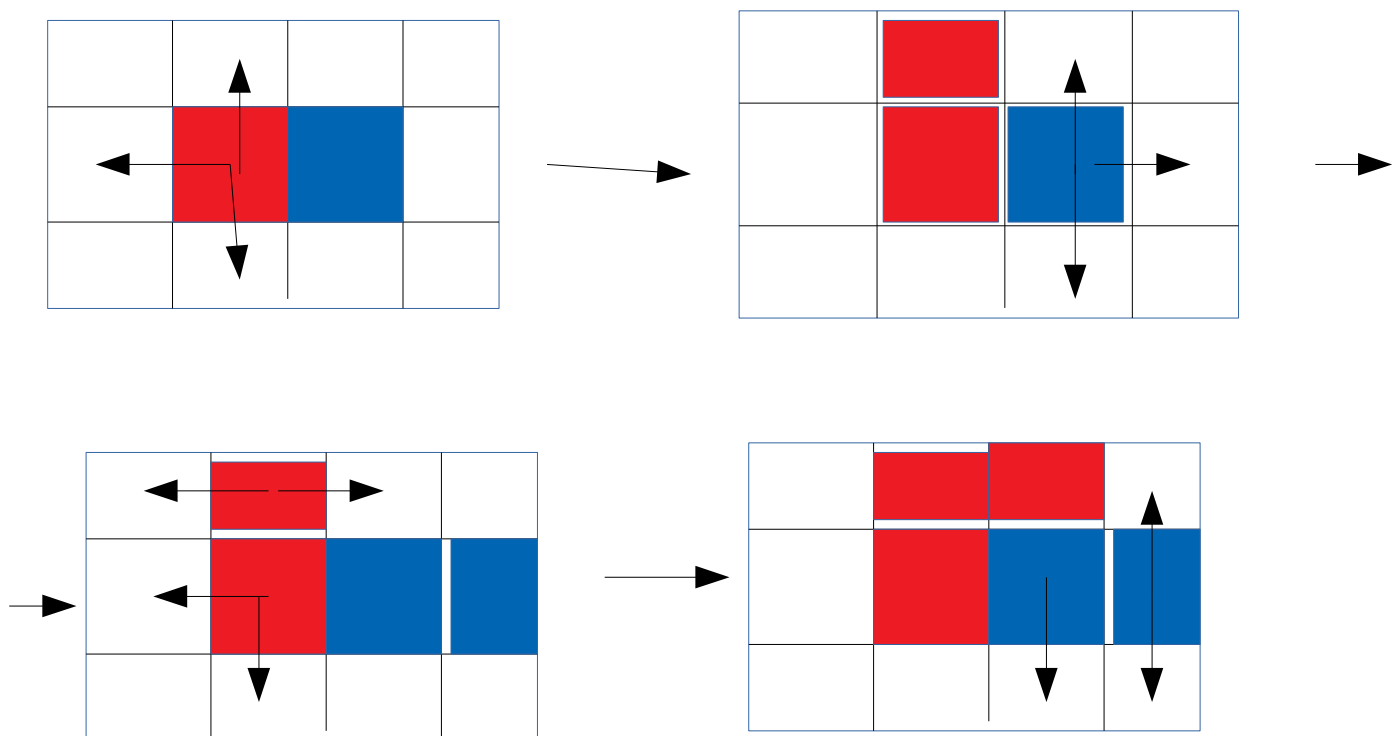
### **Generace políček defaultní hry:**

Na začátku hry se provádí generace mapy. Každé políčko mapy dostává právě jednu z (  $\text{AttrCount} \approx 10$  ) vlastností. Každou vlastnost každé políčko dostává s pravděpodobností (  $1 / \text{AttrCount}$  ).

Třeba hra může mít vodní políčko, elektrické políčko, ...

### **Tah:**

Defaultně každý hráč musí dobýt ( capture ) právě jedno políčko, které je sousední s jedním z jeho políček a zatím nikomu nepatří.



Tady červená políčka patří 1. hráči, modrá patří 2. hráči.

### **Vlastnosti políček:**

Vlastnosti políček mění, jak se hra hraje. Třeba nelze dobýt electrické políčko z vodního políčka ( z jiného sousedního políčka ale teoritické dobýt můžeme ).

### **Kde jsou začáteční políčka hráčů?:**

Každý hráč dostává 2 náhodná políčka ( všechna 4 políčka jsou různá ).  
1. hráč vybírá políčko ve kterém bude začínat. 2. hráč vidí, co si vybral 1. hráč, a potom vybírá jedno ze svých dvou políček.

### **Kdo vyhrál?**

Závěrečný počet políček je větší => výhra, stejný => remíza, menší => prohra.

### **Defaultní velká hra:**

Defaultní velká hra je posloupnost ze 2 “malých” her popsaných nahore. V 1. hře : 1. hráč hraje jako první ( dělá 1. tah ). V 2. hře : 2. hráč hraje jako první.

Výhra v “malé” hře => 1 bod, remíza => 0,5 bodů, prohra => 0 bodů. Ten, kdo má větší součet bodů na konci vyhrává, menší prohrává. Součet je stejný => remíza.

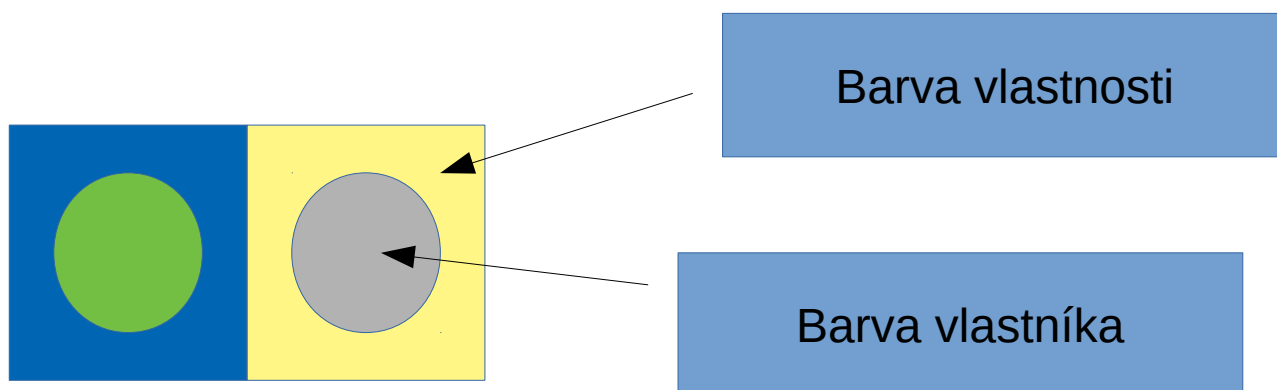
### **Hrdiny:**

Velká hra se začíná tím, že si hráče vybírají hrdinu, kterou pak budou hrát celou velkou hru. Celkem hra bude mít ~ 10 hrdin. Jak je vybírají? Každý ze dvou hráčů dostává 2 náhodné hrdiny ( všechny 4 hrdiny jsou různé ). Hráče vybírají svých hrdin bez vědomosti, co si vybrali jiné hráče ( secretly ). Až všichni hráče vyberou, pak se dozvědí, co si vybrali jiné hráče.

Různé hrdiny mají nějak změněné pravidla pro celou hru. Třeba nějaká hrdina vždy hraje jako druhá, jiná hrdina může udělat o 1 dobytí víc pro každou dvojici sousedních políček typu-(vodní,electrické), které má, apod.

//-----//

**Využití virtualizace:** generace políček, generace začátečních políček hry, generace hrdin, vizualizace hry ( ve hře bude možnost přepnutí mezi víc vzhledů : v základním vzhledu každé políčko prostě má nějakou barvu podle své vlastností + nějak označenou barvu vlastníka; ve hře bude i vzhled s obrazovky pro různé políčka ).



Nějak takhle budou vypadat políčka v základním vzhledu.

**AI řešení:** bude naimplementováno 2 typy AI : easy, medium. Easy bude vždy přijímat náhodné řešení ( ve všech fázích hry ). Medium bude používat jednoduchý BFS fixní délky s omezením časem výpočtu pro malou hru, strategie pro volbu hrdiny na začátku bude implementováno dalším způsobem: pro každé 2 různé hrdiny se provede generace  $\sim 10^5$  her ( kde budou hrát AI pomocí BFS algoritmu ), pak podle výsledku dostaneme přibližné hodnoty pravděpodobnosti výsledku velké hry mezi těmito hrdinami. Pak prostě AI bude vybírat hrdinu podle tabulky pravděpodobnosti výsledku hry ( viz základy teorií her ( Von Neumann minimax theorem )).

Výsledný kód bude mít virtualizaci i pro hráče, ale pro vytváření hry s víc než 2 hráče, kde aspoň jeden z nich je mediumAI bude potřeba napsat kód pro jeho strategii ve fázi volby hrdiny.

Samozřejmě výsledný kód bude umožňovat jednoduše měnit rozměry mřížky a počet her ve velké hře. Jedna hra nebude trvat dlouho ( aspoň ta defaultní ), takže save/load modul nebude implementován.