Programátorská dokumentace: Piškvorky s překážkami

Jedná se o klasickou hru piškvorky s lehkou modifikací: uživatel bude hrát píškvorky proti počítači, na hracím poli, kde některá políčka budou zablokována.

Hrací pole má velikost 15\*15. Uživatel bude hrát s kolečky a počítač s křížky.

Algorytmus:

„Počítač“ bude hledat tahy ve stavovém prostoru pomocí mini-max algorytmu, vylepšeným alfa-beta prořezáváním, kde vybírá tah ze sousedních políček.

Proměné

* board\_size – integer určující velikost pole (doporučuju mezi 15 a 35)
* board – 2D pole charakterů tj. Hrací plocha, čísluje se od 1 do board\_size (ve skutečnosti má okraje z „#“, takže je to pole od 0 do board\_size+1), na každém políčku je práve jedno z:
  + „X“ – kroužky
  + „O“ – kolečka
  + „#“ – překážka
  + „\_“ – prázdné políčko
* used – pole souřadnic políček obsazených kroužkama nebo křížkama
* active – pole souřadnic políček, které jsou prázdné a zároveň sousedí s obsazeným políčkem
* O\_turn=player\_starts – Booleon, True pokud zacina človek neboli „O“

Utlity

utility –funkce jenž vypocítá a vrátí score aktuální pozice na hrací desce pomocí board a used. Skóre udává jak je dáná situace výhodná pro X nebo O (0 => +- vyrovnaná pozice (score v sobe nepočítá kdo bude na tahu další), Kladné => vyhrává X, Záporné vyhrává O).

* Utility vypořítavá skóre tak, že spočítá všechny spojité řetězce X a O s cenami:
  + 1-tice: 0
  + 2-jice: 20
  + 3-jice: 60
  + 4-rice: 1000
  + 5-tice tj vyhra: 1000000.
* Dále se určuje u n-tice stav:
  + otevřený – lze rozšířit na 5-tici do obou stran \_\_\_OO\_\_
  + uzavřený – lze rozšířit na 5-tici pouze do jedné strany \_\_OOOX
  + mrtvý – NE-lze rozšířit na 5-tici XOO\_\_X
* Dále se udělují bonusy za řetězce s mezerou např: X\_X nebo OO\_O
* n-tice se od hráče X ke skóre přičítají, naopak od hráře O se odečítají
  + state\_value – vratí 5 za otevřený stav, 2 za uzavřený stav a 0 za mrtvý
  + sign – vrátí 1 pokud na vstupu byl X, -1 pokud na vstupu byl O
  + update\_state – z otevřeného stavu vrátí uzavřený a z mrtvého stavu vrátí mrtvý
  + explore\_one\_directon – prohledá v jedné polopřímce
  + explore – prohledá v jednom smeru (horizontálně, vertikálně a 2x diagonálně) tj. prohledá 2x v jedné polopřímce, určí délku řetězce, stav, přičte (popř. odečte) ke skóre hodnotu řetězce podle jeho délky a stavu a přičte (popř. odečte) bonus

Minimax

minimax – funkce jenž vrátí „nejlepší“ tah s danou hloubkou, X je MAX hráč, O je MIN hráč

* valueMIN – vrátí skóre nejlepšího tahu s danou hloubkou pro MIN hráče
* valueMAX - analogicky
* decisionMIN – vrátí nejlepšího tah s danou hloubkou pro MIN hráče
* decisionMAX - analogicky

Funkce

* show\_board – vytiskne pole board i s cislovaním sloupců a řádků
* check\_int - vrati True pokud jeji argument byl integer
* check\_split – vrati True pokud na string lze pouzit funkci split() se dvěma argumenty tj. pokud retezec lze rozdelit na právě dvě části mezerou
* check input – vrati True pokud inp je dvojice celých čísel oddělených mezerou která je mezi 1 a board\_size
* add\_to\_active\_neighbour – pro zadane policko dá všechny jeho sousedy, kteří již nejsou aktivní do pole active
* ran\_empty\_space\_middle – vrati souradnice prazneho policka +- ve prostred
* get\_mm\_active – pomocí pole mm\_used a mm\_board vytvori pole sousednich policek mm\_active a vrati ho
* minimax\_make\_move – udělá tah minimaxu s danou hloubkou pro daného hrače, zkontrokuje zda „počítač“ nevyhrál
* get\_player\_input – chce input od hrace dokud neni validni a pak vrati souradnice (dvojici intů)
* player\_make\_move – bude chtít input od hrače dokud to nebude platný tah a pak na dané políčko zahraje a zkontrokuje zda nevyhrál
* switch\_turn – obrátí hodnotu booleonu O\_turn a vrati ho
* print\_char – vytiske „O“ pokud O\_turn==True, „X“ jinak

Soubory

* board\_setup - definuje board\_size, sestaví pole board,used,active podle vstupu uživatele
* piskvorky\_console – hlavní soubor pomocí kterého se spouští celá hra
* functions – soubor různých funkcí, které se využívají napříč projektem
* minimax\_f – soubor obsahující funkcí minimax
* utility\_function - soubor obsahující funkcí utility