Dokumentace k zápočtovému programu Hra "Lodě / Battleship"

Vypracoval: Dmitry Zhukov Naposledy změněna: 18. února 2020

Zadání

Vytvořit hru "Lodě / Battleship".

Technická omezení a předpoklady

Hra nemá grafické rozhraní a vyžaduje instalaci balíčku **Python** 3 verze alespoň 3.7.5 s použitím standardní knihovny **random**. Hlavní jazyk hry je angličtina.

```
Soubory
__init__.py
agent.py
battleship.py
game.py
```

Uživatelská část

Hra se spouští příkazem python3 battleship.py.

Na začátku hry každý účastník disponuje sadou lodí: 4x ■, 3x ■■, 2x ■■■, 1x ■■■■ a 1x ■■■■.

Po spuštění se objeví okno volby režimu. (0) hráč proti hráči předpokládá reálné účastníky, (1) a (2) jsou režimy s účastí počítačového hráče.



Dále následuje okno zadávání jména hráče a volby způsobu generace mřížky.

```
Enter name of the player: Player
Auto (0) or manual (1) ship generation? 0
```

Automaticky vygenerovaná mřížka musí být potvrzena. Pokud rozestavení se nepodařilo, je třeba zadat znak <u>jiný</u> než **1** a bude možnost opětovné generace.

```
Your final grid.
               5
                  6
                        8
                              10
            4
                  0
                     0
                        O
            0
               0
                  0
                     0
                        0
                              o
6
                              0
     0
        0 0 0 0
                       0
                         0
                             0
           0
              0
                  0
                     0
Enter (1) if you agree. Else, enter any other key. 1
Player has been set. Press Enter to continue.
```

Manuální zadávání lodí probíhá volbou parametru vkládání nebo odstranění. Rozebereme vkládací režim. Hráč je vždy informován kolik zbývá vložit lodí a jakého druhu.

```
Add (a) or remove (r) ship? a

4x ■, 3x ■, 2x ■, 1x ■, 1x ■.
```

Zvolí se souřadnice začátku a konce předpokládané lodě. Poté se vytiskne výsledek operace. Loď se vkládá s příslušným diskrétním pásmem. Důvod neúspěšné operace bude krátce popsán.

Either length, shape is wrong or ships collide. Try again.

Při odstraňování lodě stačí zadat pouze jednu tečku, která jí patří. Odstraněna loď bude spolu s příslušným diskrétním pásmem.

Pokud zvolený režim předpokládá účast druhého reálného hráče, bude vyzván k zadání údajů. Pokud se předpokládá účast počítačového hráče, jeho mřížka se vygeneruje automaticky.

Po vytvoření a nastavení obou hráčů, hra zobrazí následující mřížku po každém tahu. V horní části jsou jména hráčů a statistiky zbývajících lodí. V prostřední části jsou mřížky a informace, které hráč dokázal od soupeře získat. V dolní jsou zobrazeny úspěšné kroky hráče z posledního tahu.

		_,	<u> </u>				-,			, 1x	IXCIII.				<i></i>							, 1x
	1	2	3	4	5	6	7	8	9	10			1	2	3	4	5	6	7	8	9	10
1												1										
2												2										
3												3										
4												4										
5												5										
6												6										
7												7										
8												8										
9												9										
10												10										

Příklad, na obrázku dolů je vidět, že hráč PC objevil tři lodě hráče Player a poslední úspěšná políčka jsou napsaná právě v dolní oblasti tabulky. Momentálně na tahu je hráč Player.

Playe Rem.		3x	■,	2x	■,	2x		,	1x	_	■, Ø:	·	P R		11 :	4	χΙ	,	3x	■,	2x		,	1x		, 1x
		1	2	3	4	5	6	7	8	9	10				١		1	2	3	4	5	6	7	8	9	10
	1														1										0	
	2														2										0	
	3			0											3					0					0	
	4				0										4		0	0	0	0					0	
	5							0							5		0			0	0				0	
	6														6		0	0	0	0	0				0	0
	7														7				0		0				0	
	8			0											8	3			0	0	0		0			
	9														g)										
	10														1	.0										
PC			((2,	10)	, (3,	10)	, (1,	10),	(4, 10), (5,	10),	(5	,	2),	(5	, 3),	(7,	9))		
nter co	ordi	nat	e	(ro	ı co	lum	n):																			

Hra se ukončí hláškou po vyčerpaní všech lodí jednoho z hráčů. Ukážou se zbývající pozice všech ostatních lodí.

Player Rem.	0:	0x	■,	0x	■,	0х		٠,	0x	_	, 0x	PC Rem.	7:	4x	■,	2x	■,	0x		٠,	0x	_	, 1x
	_	1	2	3	4	5	6	7	8	9	10			1	2	3	4	5	6	7	8	9	10
	1	0	0	0	0	0	0		0		0		1	0	0	0	0	0	0		0	0	
	2	0	0			0	0		0	0	0		2	0		0	0		0		0	0	
	3	0	0	0	0	0	0	0	0	0			3	0	0	0	0		0		0	0	
	4	0	0		0		0	0	0	0			4	0	0	0	0	0	0	0	0	0	
	5	0	0	0	0		0	0		0			5	0			0	0		0	0	0	
	6	0	0	0	0	0	0	0	0	0	0		6	0	0	0	0	0		0	0	0	0
	7	0						0			•		7	0	0	0		0		0		0	•
	8	0	0	0	0	0	0	0	0	0	0		8	0	0	0	0	0		0		0	0
	9	0	0	0	0	0					0		9	0		0	0	0	0	0		0	•
	10		0	0	0	0	0	0	0	0	0		16	0		0	0	0	0	0	0	0	0
PC			(2,	2),	(7	, 1	0)															

Programátorská část

Program používá různé techniky pro kontrolu správnosti zadané informace uživatelem. Nejzajímavější z hlediska implementace je generování mřížky a proces rozhodování počítačem dalšího tahu.

Automatická generace mřížky se dělá následujícím způsobem. Zvolí se políčko s · náhodně. Poté algoritmus zkusí umístit nejdelší dostupnou loď až k té nejkratší v náhodném směru. Pokud žádná dostupná nemůže být umístěná, políčko bude označeno **o** a zvolí se nové náhodné políčko.

Dále popíšeme chování počítačového hráče. Počítač rozlišuje dvě situace. Pokud předchozím krokem našel celou loď nebo teprv začíná hledat, pak vyhledá všechna políčka s · a z těch vybere náhodně. Pokud trefí do nějaké části lodě a není to letadlová loď (1), pokračuje v jednom ze 4 směrů a hledá další políčko. Vždy si pamatuje předchozí stav a pokud narazí na o, pokračuje ve vyhledávání v jiných směrech, ale v dalším tahu. Pokud objeví loď délky 2 a není to bitevní (2), dále hráč vybírá pouze ze dvou políček tak, aby loď mohl prodloužit. Směr prodloužení se vybírá náhodně. Zásadní je, že počítač vždy se bude snažit prodloužit již nalezenou část lodě, což je svým způsobem hladový algoritmus.

Kontrola zda reálný hráč objevil celou loď se dělá následujícím způsobem. Reálný hráč nemusí se chovat hladově, může střílet různě. Proto pro takového hráče se udržuje množina již objevených bodů s ■. Následujícím krokem zvolí nějaké políčko s •. Pokud objeví ■, program zjistí zbývající políčka lodě a ověří se zda jsou v množině již objevených. Pokud celá loď je v množině, počítá se jako úplně objevená. Doplní se diskrétní pásmo a všechny body lodě se z množiny odstraní.

Pro účely testování byl doplněn režim (2), umožňující spouštět hru v automatickém režimu.

Budoucí vývoj

Vývoj hry by se mohl vydat ve směru chytřejšího rozhodování počítačem (vyhledávání podmatic určité velikosti pro objevení dlouhých lodí) a celkového refaktoringu kódu. Některé funkce opakují logiku jiných ale s drobnými úpravami, což je potenciál pro zmenšení objemu zdrojového kódu.

Reference

https://cs.wikipedia.org/wiki/Lod%C4%9B