



ÚSTAV AUTOMATIZACE A INFORMATIKY

Algoritmy umělé inteligence

Jméno: Eim Jakub		ID studenta: 216797
Datum zadání: 20.3.2023	Datum odevzdání: 3.5.2023	Studijní skupina: 4pAIŘ/1
Název úlohy: Tvorba hry Snake s užitím algoritmu A*		

1 Tvorba hry Snake

Program v Pythonu se skládá celkem ze tří souborů

1.1 Snake

Jedná se o jednoduchou implementaci hry Snake v jazyce Python. Hra je implementována pomocí třídy SnakeGame, která má několik metod a vlastností.

V metodě `__init__` je nastaven počáteční stav hry. Hrací deska je reprezentována 2D polem NumPy board. Hodnota 0 v poli označuje prázdné místo, 1 reprezentuje tělo hada, 2 reprezentuje hlavu hada a -1 reprezentuje potravu, kterou had potřebuje sníst.

Počáteční pozice hada je určena náhodným výběrem vektoru rychlosti ze sady možných směrů: nahoru, dolů, doleva nebo doprava. Had má na začátku délku tří bodů a jeho tělo je reprezentováno jako seznam pozic na desce.

Metoda `update_vel` aktualizuje rychlost hada na základě uživatelského vstupu. Metoda bere jako vstup vektor rychlosti a kontroluje, zda se jedná o platný tah. Pokud se jedná o platný tah, rychlost hada se aktualizuje.

Metoda `update_state` aktualizuje stav hry na základě aktuální rychlosti a směru hada. Pokud had narazí na zeď nebo narazí do vlastního těla, hra končí. Pokud had sní potravu, jeho délka se zvýší o jedna a na desce se náhodně vygeneruje nová potravina. V opačném případě se had pohne o jednu jednotku ve směru své aktuální rychlosti.

Metoda `rand_food` generuje náhodnou pozici potravy na desce.

Nakonec je vytvořen objekt SnakeGame s výše popsanými vlastnostmi a metodami. Objekt reprezentuje aktuální stav hry a hra se hraje opakovaným voláním metody `update_state`, dokud hra neskončí.

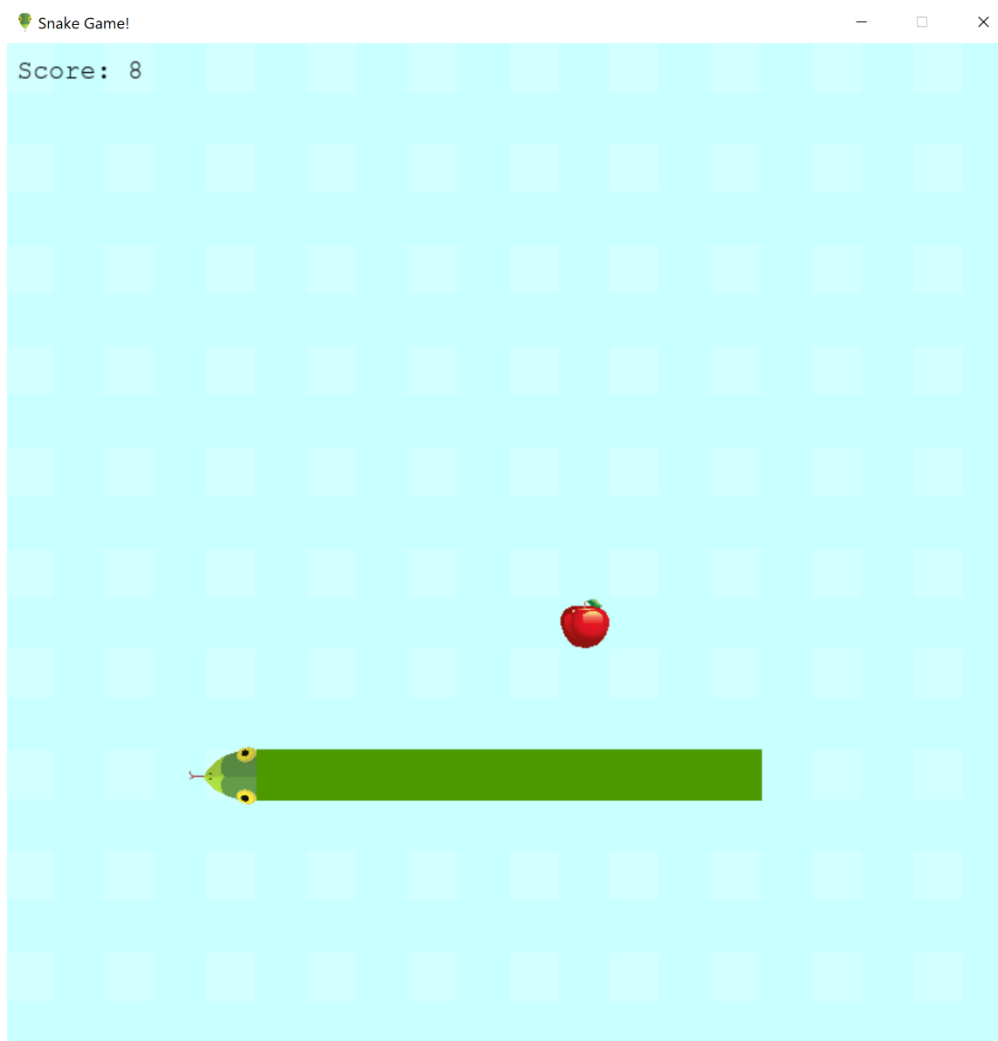
1.2 Snake_GUI

Tento kód implementuje hru Snake s grafickým rozhraním pomocí knihovny Pygame. Kromě importu modulu Pygame a několika dalších funkcí je v tomto kódu definována třída SnakeGameGUI, která dědí algoritmus hry samotné z třídy SnakeGame a přidává k ní funkce pro grafické zobrazení hry.

Metoda `__init__` třídy SnakeGameGUI inicializuje hru, nastavuje barvy a načítá obrázky pro hada a jablko a také vytvoří herní okno pomocí Pygame.

Metoda `draw_board` vykresluje hrací desku na základě aktuálního stavu hry.

Metoda `run_game` obsahuje hlavní smyčku hry a obsluhuje vstupy hráče pomocí klávesnice. Tato metoda také volá metodu `update_state` ze třídy SnakeGame a metodu `draw_board`, aby zobrazil aktuální stav hry. Je zde také ošetřeno natáčení hlavy hada s ohledem na jeho momentální rychlost a směr. Pokud dojde k ukončení hry (buď hráčem nebo prohrou), zobrazí se text "Game Over!".



Obrázek 1:1 Generované GUI

1.3 Snake_AI

Tento kód je implementací hry Snake s algoritmem A* pro hada, aby našel cestu k jídlu. Třída `SnakeGameAStar` je podtřídou třídy `SnakeGameGUI`, která poskytuje základní funkcionality hry Snake, jako je GUI, aktualizace stavu hry a samotná herní mechanika.

Třída `SnakeGameAStar` přidává metodu nalezení optimální cesty k jídlu pomocí algoritmu A*. Metoda `get_safe_moves` vrací bezpečné tahy, které jsou pro hada v jeho aktuální pozici k dispozici, zatímco metody `safe_move`, `move2food`, `wiggle2food` a `wiggle_away` vrací další tah, který had udělá na cestě k potravě.

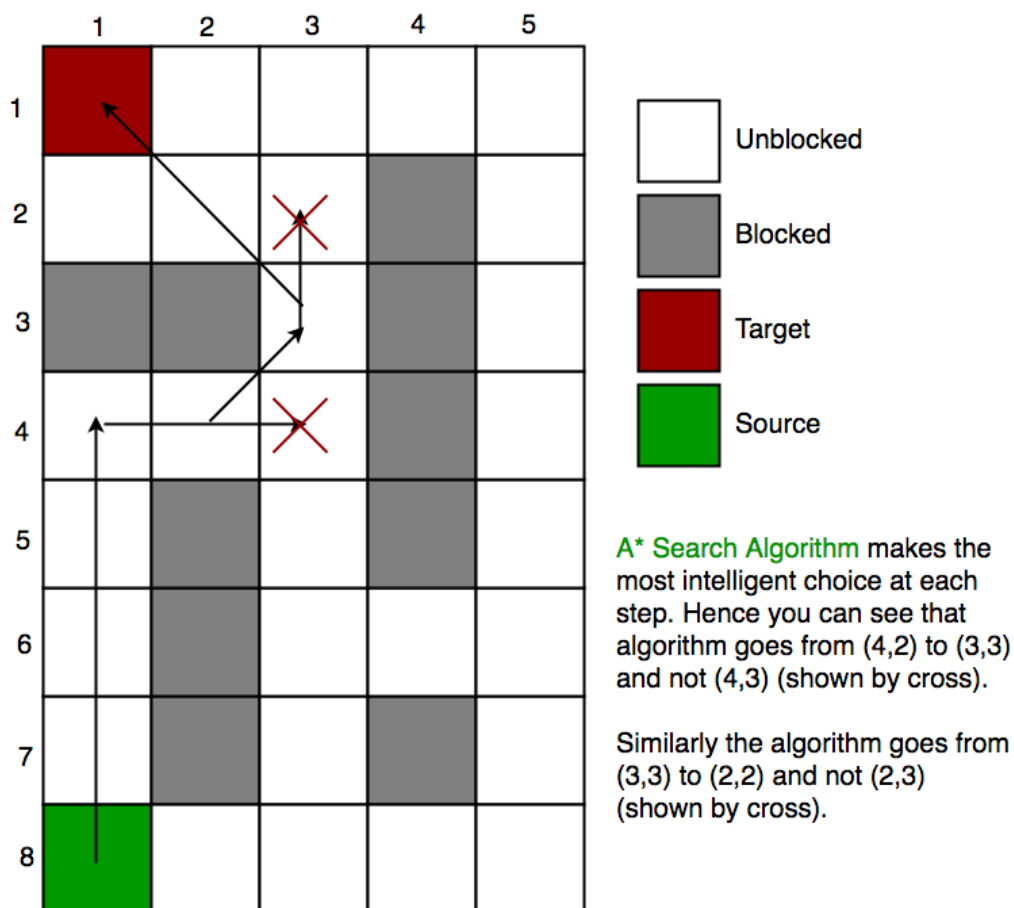
Metoda `safe_move` náhodně vybere bezpečný tah, zatímco metoda `move2food` vybere tah, který posune hada blíže k jídlu. Metoda `wiggle2food` střídavě zatáčí vlevo a vpravo, dokud nenajde tah, který vezme hada blíže k jídlu. Metoda `wiggle_away` otočí hada v směru, který ho posune co nejdál od jídla, například když cesta k jídlu v daný moment neexistuje.

Atribut `path2food` slouží k uložení optimální cesty k jídlu a vypočítá se vždy, když had sní jídlo nebo se jídlo vygeneruje na jiné místo. Pokud není atribut `path2food` prázdný, had se řídí optimální cestou k jídlu; jinak použije metodu `move2food`, aby našel další tah.

Celkově je tato implementace dobrým příkladem použití algoritmu A* k nalezení optimální cesty pro hada ve hře Snake. Nicméně může být vylepšena přidáním heuristik k algoritmu, aby byl efektivnější, například snížením prohledávaného prostoru zvažováním pouze tahů, které hada vezmou blíže k jídlu.

Při dobré generaci potravy dokáže had nahrát i skóre přes 120 bodů.

2 Princip algoritmu A*



Obrázek 2:1 Průchod polem pomocí algoritmu A*

Algoritmus A* (A-star) je jedním z nejpoužívanějších algoritmů pro hledání nejkratší cesty v grafu. Je používán v mnoha oblastech, například v robotice, videohrách a navigačních aplikacích.

Princip algoritmu A* spočívá v hledání cesty mezi dvěma body v grafu, která má nejmenší celkovou váhu. Algoritmus postupuje tím, že prochází všechny možné cesty a ohodnocuje je pomocí dvou hodnot: $g(x)$ a $h(x)$.

Hodnota $g(x)$ představuje vzdálenost od počátečního bodu k aktuálnímu bodu x a $h(x)$ představuje odhadovanou vzdálenost od bodu x k cílovému bodu. Celkové ohodnocení bodu x je poté dáno jako $f(x) = g(x) + h(x)$.

Algoritmus A* vybírá následující uzel k prozkoumání na základě hodnoty $f(x)$. Hledání cesty pokračuje, dokud není dosažen cílový bod, nebo pokud nejsou již dostupné další nezkoumané uzly.

Algoritmus A* je účinný, protože používá heuristickou funkci pro odhadnutí nejlepší cesty k cíli, ale zároveň prochází jenom ty uzly, které jsou pravděpodobně na té nejlepší cestě, což výrazně zlepšuje jeho výkon. Celkově lze říci, že algoritmus A* je výkonný a efektivní algoritmus pro hledání nejkratší cesty v grafu.

3 Zdroje

- [1] A* algorithm. *Geeksforgeeks* [online]. online: Geeksforgeeks, 2023 [cit. 2023-05-03]. Dostupné z: <https://www.geeksforgeeks.org/a-search-algorithm/>
- [2] Code a Snake Game with Python and Pygame. *YouTube* [online]. YouTube: FreeCodeCamp, 2021 [cit. 2023-05-03]. Dostupné z: <https://www.youtube.com/watch?v=8dfePIONtIs>
- [3] A* algorithm. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2023-05-03]. Dostupné z: https://en.wikipedia.org/wiki/A*_search_algorithm
- [4] *Automated Snake Game Solvers via AI Search Algorithms*. University of California, 2022. Research. University of California. Vedoucí práce Joan Aguilar Mayans, Shu Kong.