

Lights Out

V našom zadaní máme implementované riešenie hry Lights Out pomocou 3 vyhľadávacích algoritmov.

Reprezentácia stavu hry

Keďže v nasledujúcich algoritmoch sa vyžaduje vyhľadávanie v stavovom priestore (grafe), pokladali sme za potrebné definovať si štruktúru Node, ktorá obsahuje:

- Stav hry (Rozpoloženie zasvietených políčok)
- Odkaz na rodiča (Stav, z ktorého sme dostali aktuálny stav)
- Akciu, ktorá bola vykonaná na rodičovskom stave
- $h(n)$ - Hodnotu heuristickej funkcie
- $g(n)$ - Hodnotu cenovej funkcie (vzdialenosť od počiatočného stavu)

DFS

Algoritmus DFS máme implementovaný iteratívne, zásobník je reprezentovaný pomocou dátovej štruktúry **stack**.

DFS spočíva v brute-force prístupe, ktorý prechádza stavový priestor (graf) nasledovným spôsobom:

1. Vyberie si posledne pridaný stav zo zásobníka
2. Ak stav ešte nebol expandovaný, expanduje ho a označí ako už expandovaný
 - Ak susedný stav, ktorý algoritmus našiel expandovaním predstavuje finálny stav hry, algoritmus našiel riešenie

Greedy

Algoritmus Greedy máme implementovaný iteratívne, zásobník je reprezentovaný pomocou dátovej štruktúry **priority queue**, ktorú zoradíme na základe heuristickej funkcie $h(n)$.

Heuristická funkcia vyzerá nasledovne:

$$h(n) = \text{Pocet zasvietenych policok}$$

Algoritmus prechádza stavový priestor (graf) nasledovným spôsobom:

1. Vyberie stav s minimálnou hodnotou $h(n)$ zo zásobníka
2. Ak stav predstavuje finálny stav hry, algoritmus našiel riešenie
3. Ak stav ešte nebol expandovaný, expanduje ho a označí ako už expandovaný

A*

Algoritmus A* máme implementovaný iteratívne, zásobník je reprezentovaný pomocou dátovej štruktúry **priority queue**, ktorú zoradíme na základe funkcie $f(n)$.

Funkcia $f(n)$ vyzerá nasledovne:

$$f(n) = h(n) + g(n)$$

$$f(n) = \text{Pocet zasvietenych policok} + \text{vzdialenost od pociatocneho stavu}$$

Algoritmus prechádza stavový priestor (graf) nasledovným spôsobom:

1. Vyberie stav s minimálnou hodnotou $f(n)$ zo zásobníka
2. Ak stav predstavuje finálny stav hry, algoritmus našiel riešenie
3. Ak stav ešte nebol expandovaný, expanduje ho a označí ako už expandovaný

DFS

Stav	Čas (s)	Dĺžka riešenia	Expandované uzly	Objavené uzly
5x5 - 1	0.0031	1	1	20
5x5 - 3	2.3528	49	49	1220
5x5 - 4	16.5933	120	120	2983
5x5 - 5	323.6797	501	501	12509
5x5 - 6	12.0168	106	106	2630
5x5 - 6	56.1159	218	218	5442
5x5 - 7	43.3203	181	181	4515
2x3 - 1	0.0005	1	1	6
2x3 - 2	0.0023	2	2	11
2x3 - 3	0.0042	3	3	16

Greedy

Stav	Čas (s)	Dĺžka riešenia	Expandované uzly	Objavené uzly
5x5 - 1	0.0032	1	1	26
5x5 - 3	0.1072	9	9	226
5x5 - 4	0.0156	4	4	101
5x5 - 5	1.8582	31	52	1301
5x5 - 6	1.7792	32	45	1126
5x5 - 6	148.8799	138	470	11751
5x5 - 7	5.323	39	115	2901
2x3 - 1	0.001	1	1	7
2x3 - 2	0.0025	2	2	13
2x3 - 3	0.0092	5	5	31

A*

Stav	Čas (s)	Dĺžka riešenia	Expandované uzly	Objavené uzly
5x5 - 1	0.0033	1	1	26
5x5 - 3	0.0436	3	7	176
5x5 - 4	0.0201	4	4	101
5x5 - 5	0.0643	5	11	276
5x5 - 6	0.1905	6	24	601
5x5 - 6	0.4376	6	63	1576
5x5 - 7	3.3921	7	360	9001
2x3 - 1	0.001	1	1	7
2x3 - 2	0.0017	2	2	13
2x3 - 3	0.0072	3	8	49

Čas (s)

Stav	DFS	Greedy	A*
5x5 - 1	0.0031	0.0032	0.0033
5x5 - 3	2.3528	0.1072	0.0436
5x5 - 4	16.5933	0.0156	0.0201
5x5 - 5	323.6797	1.8582	0.0643
5x5 - 6	12.0168	1.7792	0.1905
5x5 - 6	56.1159	148.8799	0.4376
5x5 - 7	43.3203	5.323	3.3921
2x3 - 1	0.0005	0.001	0.001
2x3 - 2	0.0023	0.0025	0.0017
2x3 - 3	0.0042	0.0092	0.0072

Počet krokov riešenia

Stav	DFS	Greedy	A*
5x5 - 1	1	1	1
5x5 - 3	49	9	3
5x5 - 4	120	4	4
5x5 - 5	501	31	5
5x5 - 6	106	32	6
5x5 - 6	218	138	6
5x5 - 7	181	39	7
2x3 - 1	1	1	1
2x3 - 2	2	2	2
2x3 - 3	3	5	3

Počet expandovaných uzlov

Stav	DFS	Greedy	A*
5x5 - 1	1	1	1
5x5 - 3	49	9	7
5x5 - 4	120	4	4
5x5 - 5	501	52	11
5x5 - 6	106	45	24
5x5 - 6	218	470	63
5x5 - 7	181	115	360
2x3 - 1	1	1	1
2x3 - 2	2	2	2
2x3 - 3	3	5	8

Počet objavených uzlov

Stav	DFS	Greedy	A*
5x5 - 1	20	26	26
5x5 - 3	1220	226	176
5x5 - 4	2983	101	101
5x5 - 5	12509	1301	276
5x5 - 6	2630	1126	601
5x5 - 6	5442	11751	1576
5x5 - 7	4515	2901	9001
2x3 - 1	6	7	7
2x3 - 2	11	13	13
2x3 - 3	16	31	49