

UMELÁ INTELIGENCIA - CVIČENIA 2

Autor: Marián Kravec

Úloha 1

Algoritmus	Časová zložitosť	Priestorová zložitosť	Pár viet
Breadth-First Search (BFS)	$O(b^d)$	$O(b^d)$	(d - hĺbka najplytkejšieho cieľa) Algoritmus prechádza strom po úrovniach. Jeho nevýhodou je pomerne veľké množstvo pamäte keďže si musí pamätať celé úrovne ktorých veľkosť môže rásť exponenciálne
Depth-First Search (DFS)	$O(b^m)$	$O(bm)$	(m - hĺbka stromu) Algoritmus prechádza strom do hĺbky, čiže kým sa z daného vrcholu dá pokračovať nižšie tak tam pokračuje. Nevýhodou je, že pri hlbokých stromoch môže tento algoritmus tráviť veľké množstvo času hlboko v strome aj napriek tomu, že možné riešenie je vo vrchných vrstvách inej vetvy
Depth-Limited Search (DLS)	$O(b^l)$	$O(bl)$	Modifikácia DFS algoritmu, kde má algoritmus povedanú maximálnu hĺbku l do ktorej sa má vnárať. Výhodou je lepší čas ako DFS avšak nevýhodou je, že nájdenie riešenia nie je zaručené (a to ani v prípade, že vieme, že niekde v strome riešenie je)
Iterative Deepening Depth-First Search (IDDFS)	$O(b^d)$	$O(bd)$	Modifikácia DLS algoritmu kde sa hodnota limitu l postupne zvyšuje, kým algoritmus nenájde riešenie. Týmto rieši problém DFS, že sa môže dlho pohybovať na nevhodnej vetve (vďaka DLS) a zároveň rieši problém DLS, že nemusí nájsť riešenie postupným zväčšovaním hĺbky
Uniform Cost Search (UCS)	$O(b^{1+\lceil l/e \rceil})$	$O(b^{1+\lceil l/e \rceil})$	(l - dĺžka najkratšej cesty, e - minimálna cena hrany) V tomto algoritme vkladáme nové vrcholy do prioritnej fronty na základe ceny. Ide o variant Dijkstrovho algoritmu
Best-First Search (BeFS)	$O(b^m)$	$O(b^m)$	Tento algoritmus na základe heuristiky určí pre každý nový očakávanú cenu do cieľa a následne podobne ako UCS ho vložíme do prioritnej fronty z ktorej následne vyberáme, ďalší prehľadávaný vrchol
A*	$O(b^d)$	$O(b^d)$	Ide o rozšírenie BeFS (v podstate pridanie UCS) kde keď nové vrcholy pridávame do prioritnej fronty na základe súčty ich doterajšej ceny a ceny ktorú predpokladá heuristika do cieľa