

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH**

—o0o—



**BÁO CÁO MÔN TRÍ TUỆ NHÂN TẠO
BÀI TẬP 01: DFS/BFS/UCS FOR SOKOBAN**

Giảng viên: Lương Ngọc Hoàng
Sinh viên thực hiện: Nguyễn Gia Bảo - 22520108

Thành phố Hồ Chí Minh, ngày 3 tháng 4 năm 2024

1 Mô hình hoá Sokoban

- **Trạng thái khởi đầu:** là trạng thái ban đầu của bản đồ, chứa các vị trí trống, các ô chứa bức tường, tất cả các hộp, các ô đích và vị trí của người chơi.
- **Trạng thái kết thúc:** là trạng thái mà vị trí mỗi hộp đều đang trùng với một vị trí đích (tất cả các ô đích đều được lấp đầy bởi hộp).
- **Không gian trạng thái:** là tập hợp tất cả các trạng thái có thể xảy ra của bản đồ, mỗi trạng thái là một cách bố trí các bức tường, các vị trí trống, các hộp, các đích và vị trí người chơi.
- **Các hành động hợp lệ:** là những hành động mà người chơi có thể thực hiện là di chuyển theo các hướng: lên, xuống, qua trái, qua phải và hành động đẩy các hộp theo những hướng trên.
- **Hàm tiến triển (successor function):** là hàm tạo ra một trạng thái mới từ trạng thái hiện tại sau khi thực hiện một hành động hợp lệ.

2 Thống kê độ dài đường đi

Bản Đồ	DFS	BFS	UCS
1	79	12	12
2	24	9	9
3	403	15	15
4	27	7	7
5	*	20	20
6	55	19	19
7	707	21	21
8	323	97	97
9	74	8	8
10	37	33	33
11	36	34	34
12	109	23	23
13	185	31	31
14	865	23	23
15	291	105	105
16	*	34	34
17	Không có lời giải		
18	*	*	*

(*) Không thể chạy được màn chơi trong vòng 10 phút.

Nhận xét:

- **DFS:** Lời giải của DFS có số bước đi lớn nhất và lớn hơn nhiều so với BFS và UCS. Bên cạnh đó, DFS còn tốn nhiều RAM bởi vì thuật toán đi sâu xuống để tìm kiếm lời giải, bản đồ càng lớn thì khả năng giải được của DFS càng giảm.
- **BFS:** Luôn tìm được lời giải tối ưu của bài toán nhưng thường chậm hơn DFS ở các bản đồ vừa và nhỏ. Trường hợp xấu nhất của BFS là khi lời giải nằm ở node lá cuối cùng của cây. Khi đó, BFS tốn rất nhiều bộ nhớ vì phải lưu trữ một lượng lớn các node. Vì vậy, ở bản đồ 5 mặc dù con người có thể giải rất nhanh nhưng BFS lại tìm ra lời giải rất chậm (vì cái cây bùng ra là rất lớn).
- **UCS:** Lời giải của UCS là tối ưu và có số bước đi bằng với BFS. Hiệu suất của UCS phụ thuộc vào hàm cost. Hàm cost càng tốt thì UCS sẽ giải càng nhanh và tiết kiệm bộ nhớ do chỉ đi vào hướng mang lại kết quả tốt nhất. Cụ thể là trong bản đồ 5, UCS tìm ra lời giải khá nhanh, còn BFS tìm ra lời giải rất lâu, DFS thì không chạy được.
- So sánh cả 3 thuật toán áp dụng cho Sokoban, ta thấy UCS tốt hơn cả, bởi vì nó tìm ra được lời giải tối ưu, tốn ít chi phí về thời gian và bộ nhớ.
- Trong tất cả các bản đồ có sẵn thì bản đồ khó giải nhất là bản đồ 18. Bởi vì nó có nhiều thùng, khiến cho không gian trạng thái lớn, dẫn đến việc tìm ra lời giải rất tốn kém về mặt thời gian và bộ nhớ.