

BÀI TẬP TUẦN 8

Bài tập 1: Cài đặt **Uniform Cost Search** để giải 8-puzzle với start state và goal state như hình bên dưới trong trường hợp chi phí được tính như sau:

Actions: dịch chuyển up/down thì cost =2;

Actions: dịch chuyển left/right thì cost =1

Yêu cầu: sử dụng ngôn ngữ Python, solution ghi dưới dạng chuỗi, ví dụ (optimal solution của problem này: ['DOWN', 'RIGHT', 'UP', 'LEFT', 'DOWN', 'LEFT', 'UP', 'UP']) hoặc xuất dạng đồ họa (animation).

2	8	3
1	6	4
7		5

Start State

1	2	3
8		4
7	6	5

Goal State

Gợi ý:

- Chứa state: có thể dùng `list`, hoặc 1D `numpy array`.
- Tập biên: có thể dùng `collections.deque` hoặc `queue.LifoQueue`.
- Animation: có thể dùng `matplotlib.animation`
- Giao diện đồ họa: có thể dùng `tkinter` (có thể dùng với QtDesigner).
- Khuyến khích thêm các chức năng như GUI, animation...

Lưu ý:

- Nếu tham khảo code của người khác thì **cần phải**:
 1. **Ghi rõ nguồn** trong file code, ví dụ `# Nguồn phần giao diện: https://...`
 2. **Đọc hiểu** code được dùng, thể hiện bằng các comments giải thích của bạn.
 3. **Có đóng góp**, cải tiến, ví dụ thêm GUI, animation nếu code tham khảo chưa có.
- **Có thể làm cá nhân hoặc làm nhóm**. Nếu làm nhóm thì trong file code cần có comment ghi **MSSV, họ tên** các thành viên THAM GIA LÀM (thực làm chứ không cần ghi hết nhóm cuối kỳ). Khi nộp bài chỉ cần một thành viên nhóm đại diện nộp.