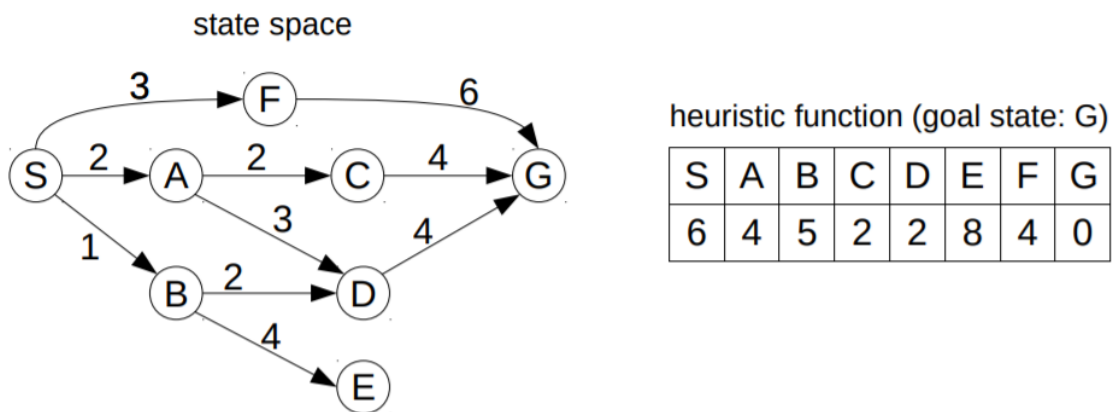


Câu 1:

Cho đồ thị và heuristic như sau:



Tìm đường đi từ S đến G bằng:

- BFS
- DFS
- Greedy với h bên trên
- A* với h bên trên

Ứng với từng thuật toán, vẽ câu tìm kiếm thể hiện quá trình mở rộng các node và các chi phí tương ứng.

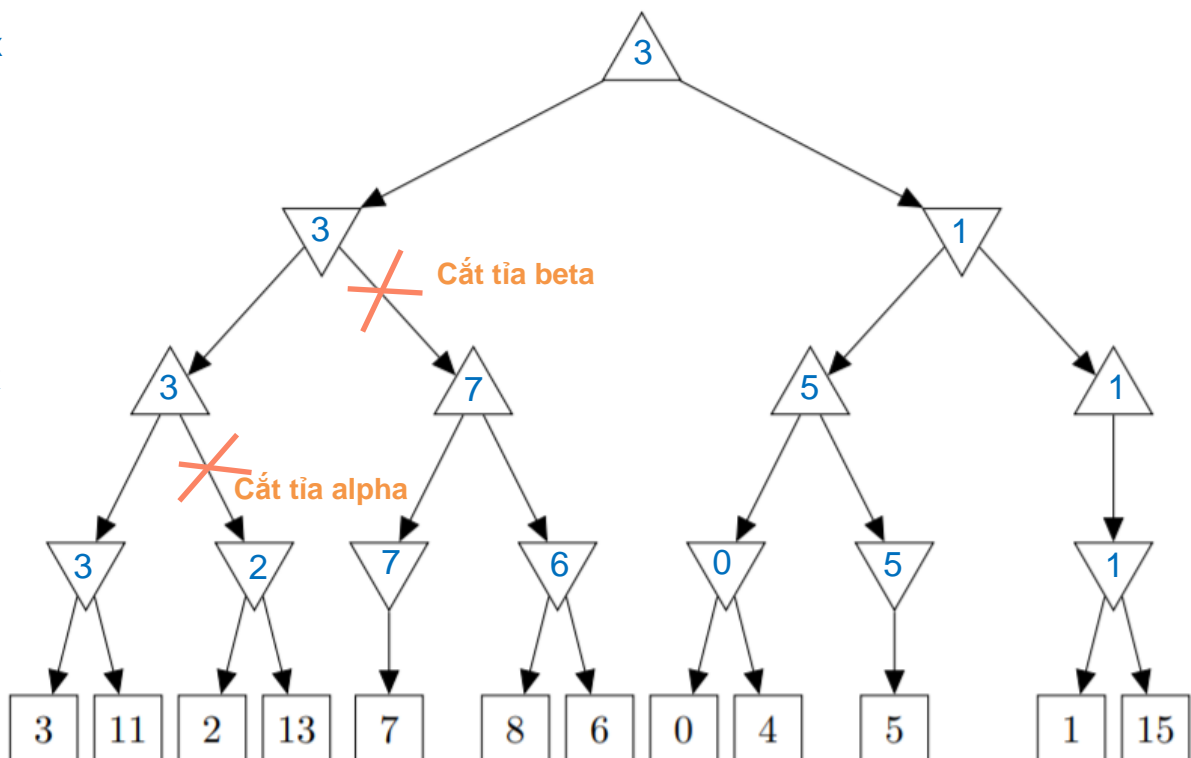
Câu 2

Max

Min

Max

Min

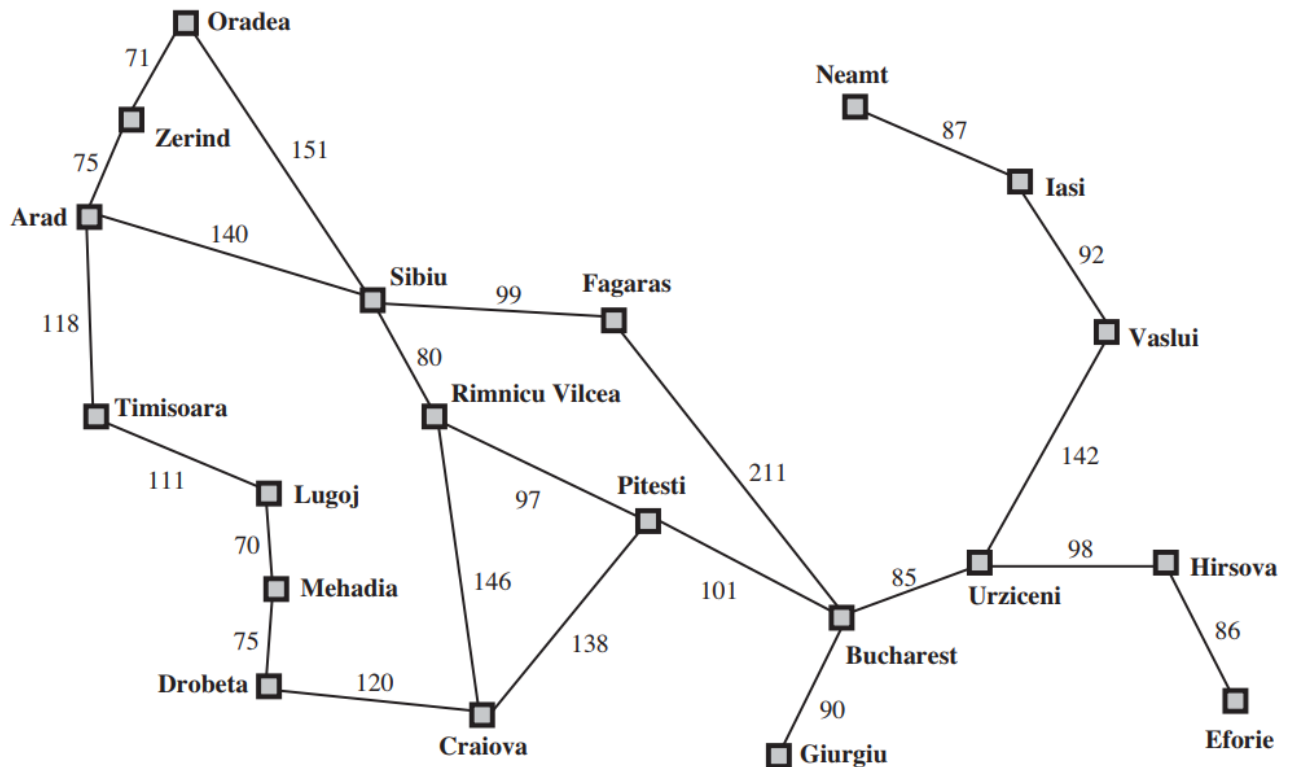


- a. Biết rằng tam giác **hướng lên** trên là **người chơi**, tam giác hướng **xuống** dưới là **đội thủ**. Hãy sử dụng thuật toán **Minimax** để chọn chiến lược cho từng lượt của hai bên.

- b. Sử dụng phương pháp **tỉa nhánh alpha-beta** để tỉa các nhánh trong cây trên. Giải thích ngắn gọn lý do tỉa

Câu 3

Giả sử rằng có hai người bạn sống ở hai thành phố khác nhau trên bản đồ, ví dụ như bản đồ dưới đây.



Tại mỗi lượt đi, hai người có thể di chuyển đến các thành phố láng giềng tương ứng của họ. Thời gian di chuyển từ thành phố i đến thành phố j bằng khoảng cách giữa 2 thành phố $d(i, j)$. Tại mỗi lượt, mỗi người chỉ có thể di chuyển một lần. Nếu một người đến trước, họ cũng phải đợi người khi hoàn xong thành xong việc di chuyển trước khi bắt đầu một lượt mới. Ta muốn hai người này gặp nhau càng sớm càng tốt.

- a. Hãy định nghĩa bài toán tìm kiếm này. Cụ thể hãy cho biết.
- Không gian trạng thái. Kích thước của không gian trạng thái.
 - Trạng thái bắt đầu.
 - Hàm kiểm tra trạng thái đích
 - Hành động
 - Hàm “successor”
 - Hàm tính chi phí
- b. Gọi $d(i, j)$ là khoảng cách đường chim bay từ thành phố i đến thành phố j . Các heuristic nào sau đây là hợp lệ (admissible):
- $D(i, j)$
 - $2 * D(i, j)$
 - $D(i, j)/2$
- c. Có bản đồ nào với tất cả các thành phố đều liên thông nhưng không tồn tại lời giải hay không.

- d. Có bản đồ nào mà tất cả lời giải đòi hỏi một người phải viếng thăm một thành phố hai lần không?

Câu 4

Trong trò chơi 8-puzzle, cho trạng thái start và trạng thái goal như sau:

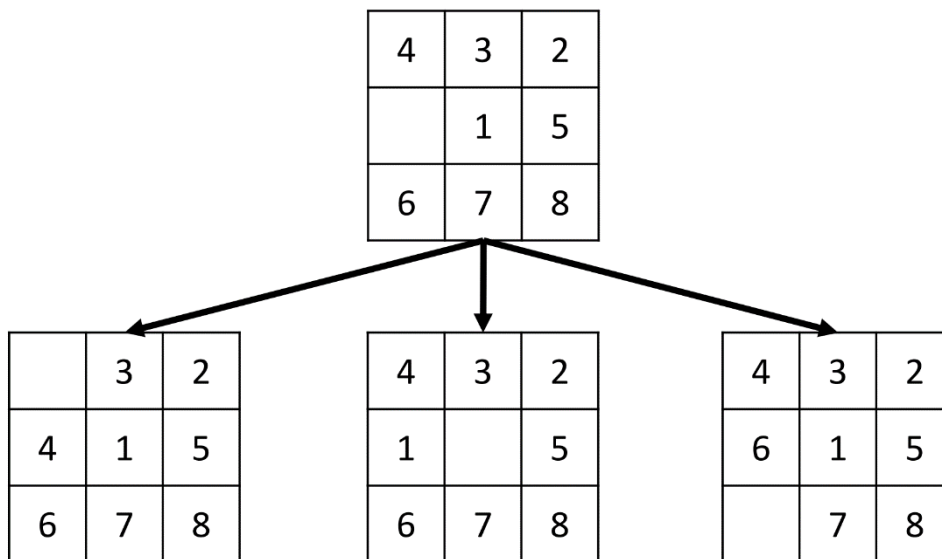
4	3	2
	1	5
6	7	8

start

	1	2
3	4	5
6	7	8

goal

Luật chơi: từ một trạng thái, bạn có thể đẩy một ô số sang ô bên cạnh với điều kiện ô bên cạnh là ô trống (không được đẩy chéo). Ví dụ, với trạng thái start ở trên, sẽ có 3 trạng thái kế tiếp có thể có:

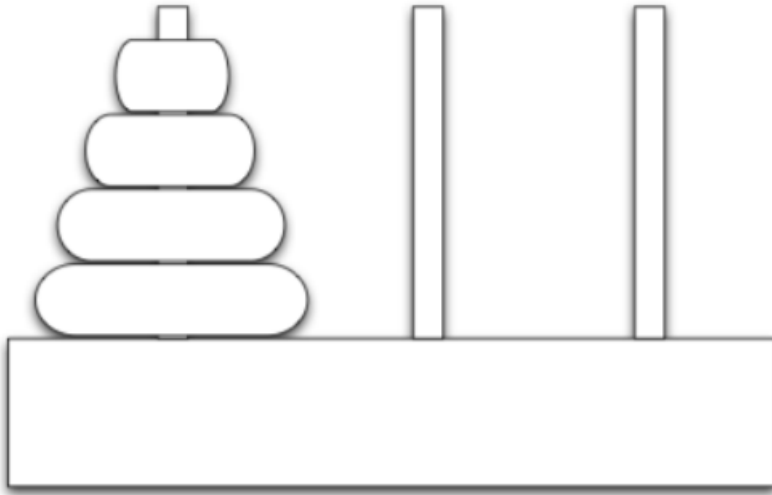


Hãy chạy thuật toán A* (sử dụng closed set) để tìm ra cách đẩy từ trạng thái start sang trạng thái goal. Ở đây, chi phí của mỗi lần đẩy đều bằng nhau và bằng 1. Hàm heuristic h mà bạn sẽ sử dụng là tổng khoảng cách Manhattan từ vị trí mỗi ô số trong trạng thái đang xét đến vị trí đúng (vị trí trong trạng thái goal). Ví dụ, với trạng thái start ở trên:

- Khoảng cách Manhattan từ vị trí ô (có) số 1 đến vị trí đúng bằng 1.
- Khoảng cách Manhattan từ vị trí ô số 2 đến vị trí đúng bằng 0.
- Khoảng cách Manhattan từ vị trí ô số 3 đến vị trí đúng bằng 2.
- Khoảng cách Manhattan từ vị trí ô số 4 đến vị trí đúng bằng 2.
- Bốn ô số còn lại đều đã đúng vị trí nên sẽ có khoảng cách Manhattan bằng 0.

Vậy: $h(\text{start}) = 1 + 0 + 2 + 2 + 0 + 0 + 0 + 0 = 5$.

Bài 5



Bài toán tháp Hà nội. Đây là một trò chơi toán học gồm 3 cột và số đĩa nhiều hơn 1. Với quy tắc các đĩa nhỏ phải nằm trên các đĩa lớn. Nhiệm vụ của trò chơi là di chuyển các đĩa từ cột ban đầu (cột trái) sang cột khác (cột phải), có thể sử dụng cột trung gian (cột giữa) sao cho vẫn đảm bảo thứ tự ban đầu của các đĩa: đĩa nhỏ nằm trên đĩa lớn.

Đây là bài toán thường được dùng để minh họa ý tưởng cho các thuật toán đệ quy và toán rời rạc.

Hãy định nghĩa bài toán tìm kiếm này. Cụ thể hãy cho biết.

- Không gian trạng thái. Kích thước của không gian trạng thái.
- Trạng thái bắt đầu.
- Hàm kiểm tra trạng thái đích
- Hành động
- Hàm “successor”
- Hàm tính chi phí