

CƠ SỞ TRÍ TUỆ NHÂN TẠO

ĐỒ ÁN THUẬT TOÁN TÌM KIẾM



Bộ môn Khoa học máy tính

Khoa Công nghệ thông tin

Trường Đại học Khoa học tự nhiên, ĐHQG-HCM

MỤC LỤC

1	Danh sách thành viên	3
2	Chạy thử trên các bản đồ.....	4
	Bản đồ 1	4
	Bản đồ 2	11
	Bản đồ 3	17
	Bản đồ 4	22
	Bản đồ 5	28
3	Nhận xét chung trên bản đồ không điểm thưởng.....	33
	Đánh giá trên từng thuật toán.....	33
	Đánh giá 2 hàm heuristic	34
4	Tài liệu tham khảo	35

1

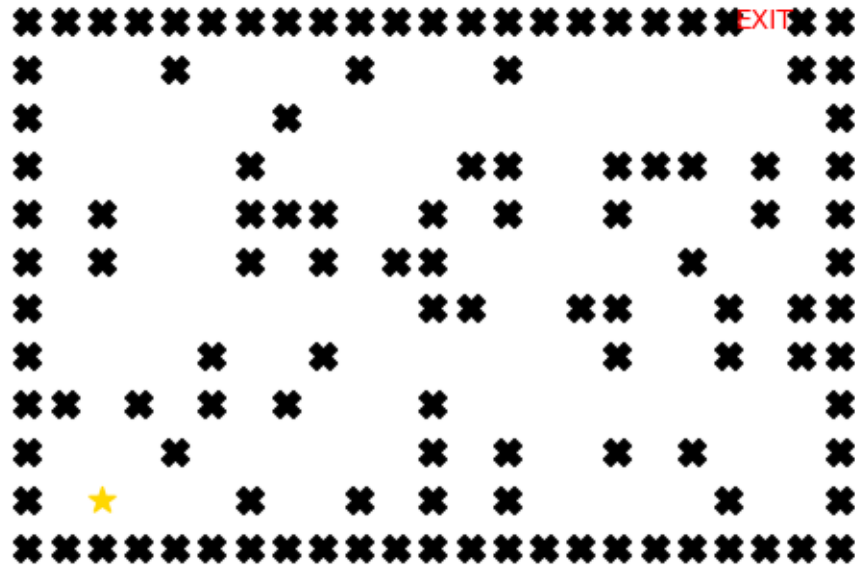
Danh sách thành viên

MSSV	Họ tên	Email
19120207	Hồ Hoàng Duy	19120207@student.hcmus.edu.vn
19120252	Hà Bảo Khang	19120252@student.hcmus.edu.vn

2

Chạy thử trên các bản đồ

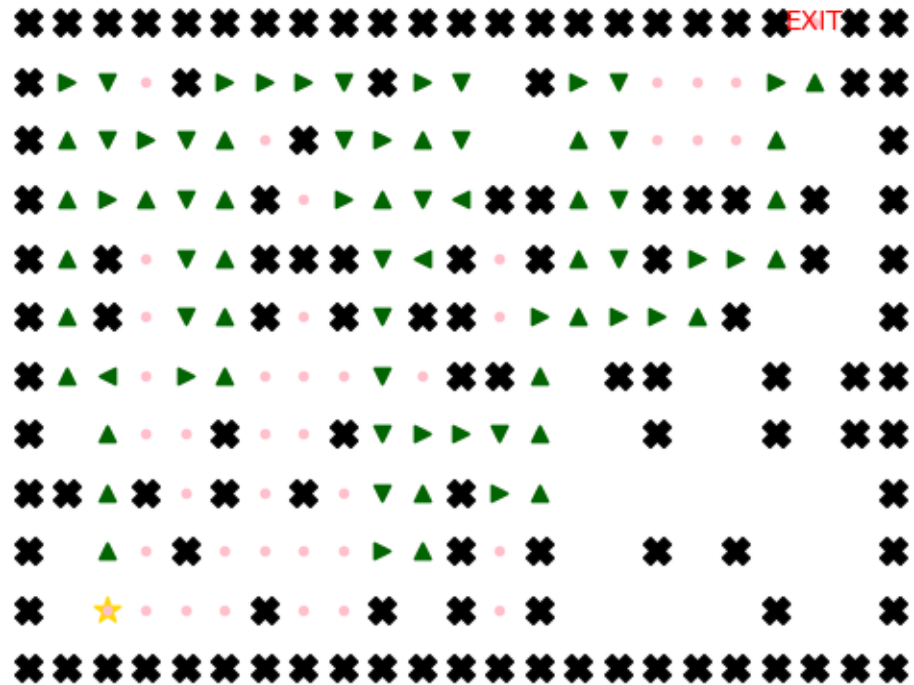
Bản đồ 1



Điểm bắt đầu $(x, y) = (10, 2)$

Điểm kết thúc $(x, y) = (0, 20)$

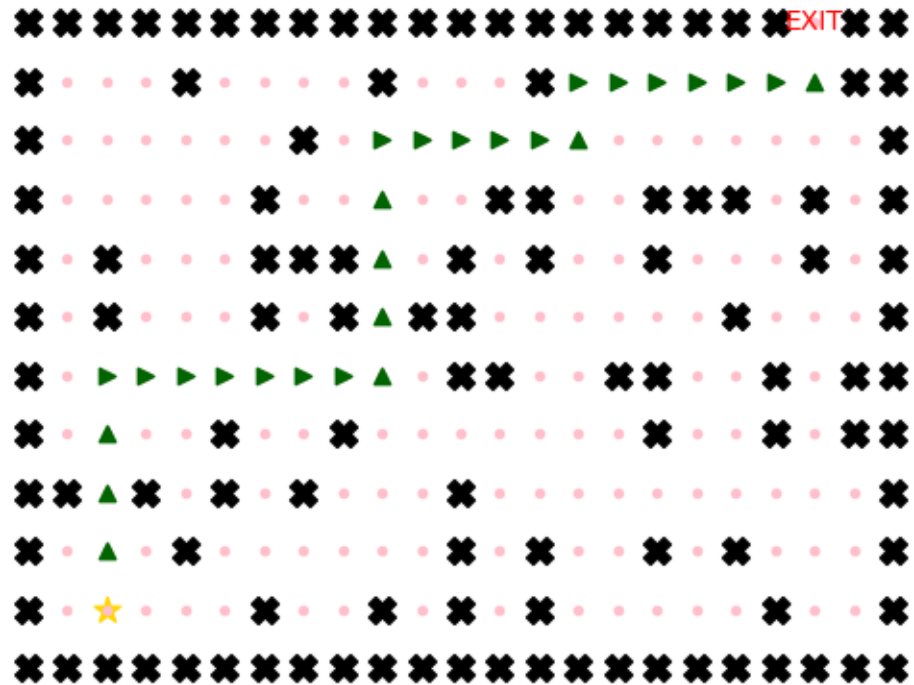
Thuật toán DFS



Độ dài đường đi: 77

Tổng chi phí: 115

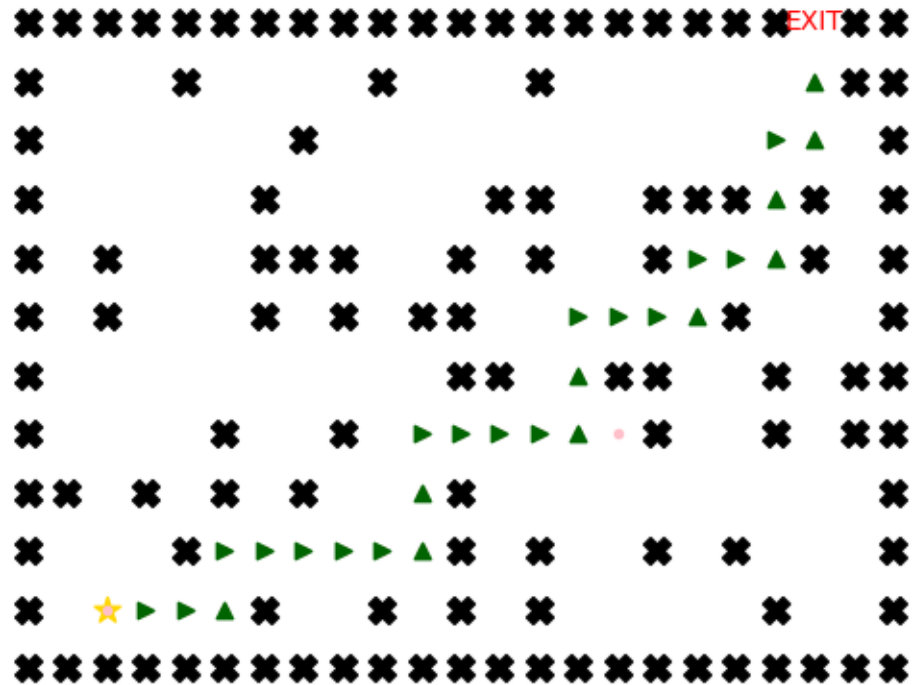
Thuật toán BFS



Độ dài đường đi: 29

Tổng chi phí: 159

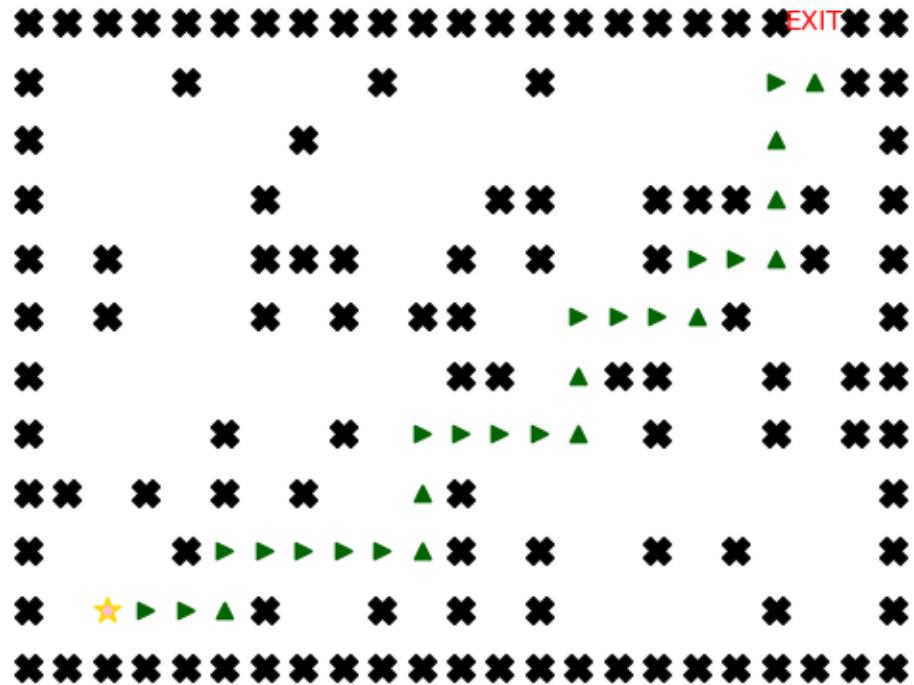
Thuật toán GBFS Manhattan



Độ dài đường đi: 29

Tổng chi phí: 29

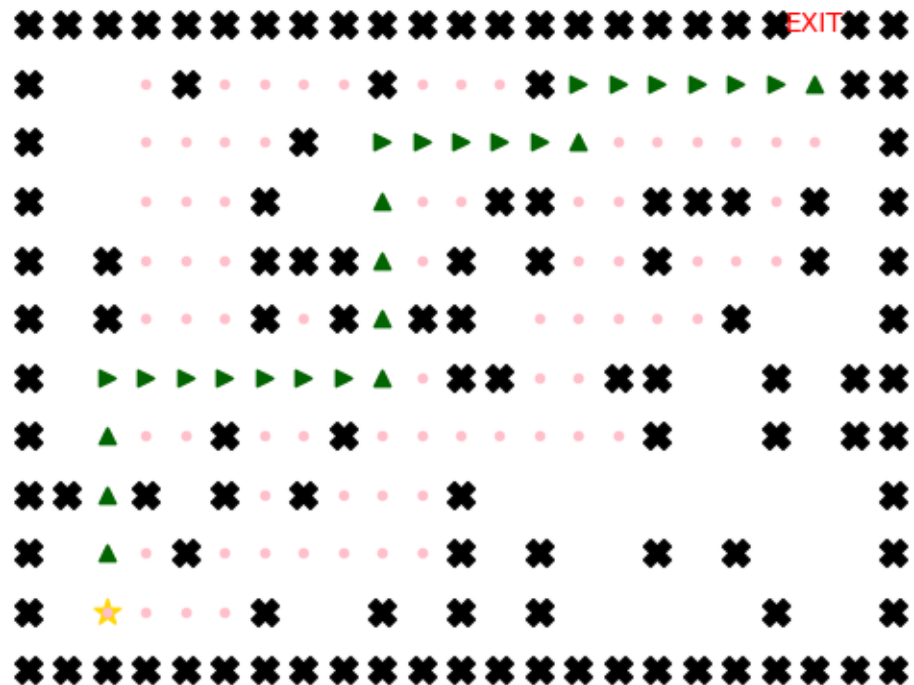
Thuật toán GBFS Euclid



Độ dài đường đi: 29

Tổng chi phí: 28

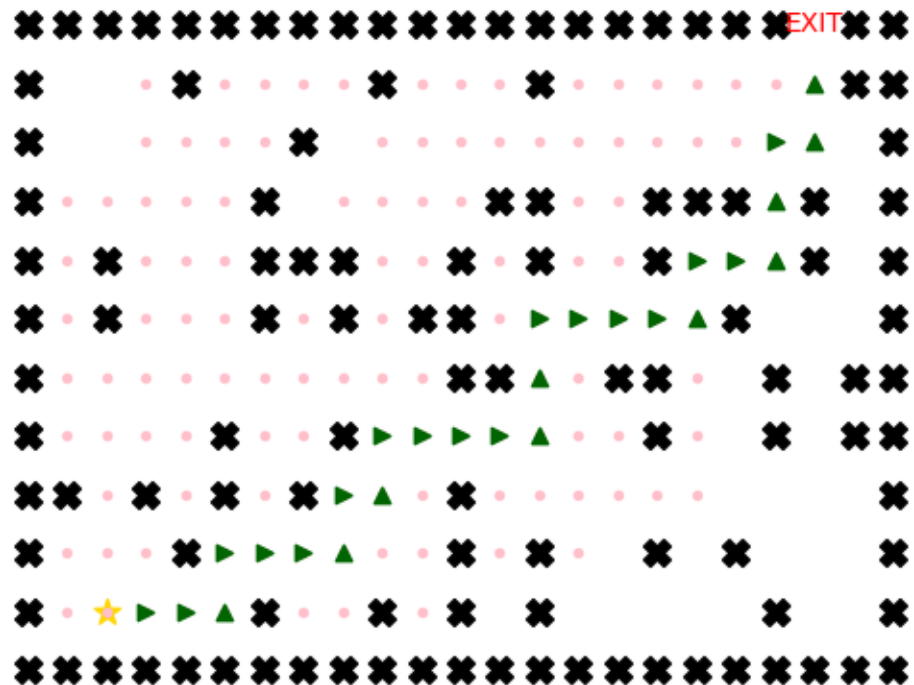
Thuật toán A* Manhattan



Độ dài đường đi: 29

Tổng chi phí: 100

Thuật toán A* Euclid

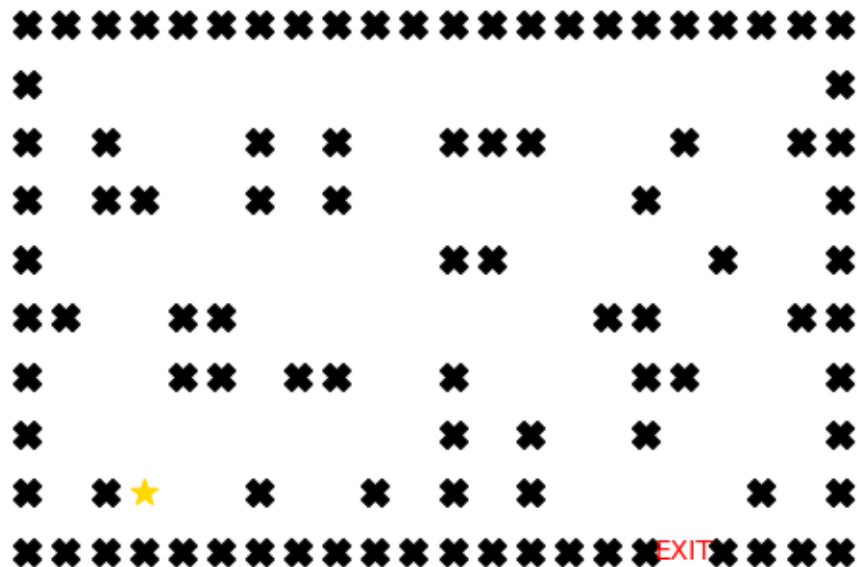


Độ dài đường đi: 29

Tổng chi phí: 125

Nhận xét

- Đối với bản đồ 1, sử dụng thuật toán GBFS với heuristic là Euclid, ta thu được đường đi tốt nhất có độ dài là 29, chi phí duyệt không thừa bất kiểm nào trên bản đồ (chi phí: 28).
- A*, BFS trong trường hợp này cũng đưa ra được đường đi tối ưu, nhưng chi phí bỏ ra để duyệt là quá lớn cho nên chúng chỉ bảo đảm về mặt kết quả chứ không bảo đảm về mặt chi phí.
- Kết quả của GBFS cũng có thể cho là một kết quả tốt, tuy nhiên đường đi lại chưa tối ưu.
- DFS cho ta kết quả tệ nhất, chi phí cũng chưa được bảo đảm nhưng vẫn tốt hơn A* và BFS.

Bản đồ 2

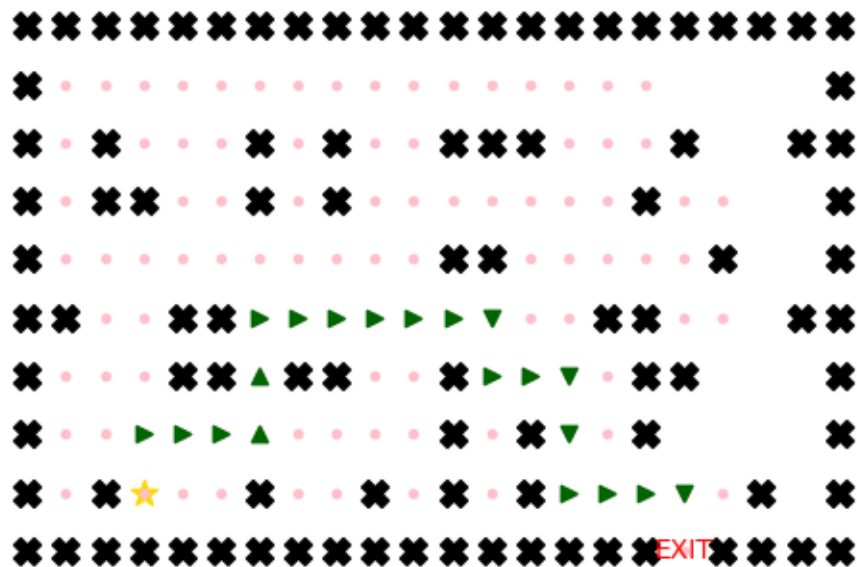
Điểm bắt đầu $(x, y) = (8, 3)$

Điểm kết thúc $(x, y) = (9, 17)$

Thuật toán DFS

Độ dài đường đi: 62

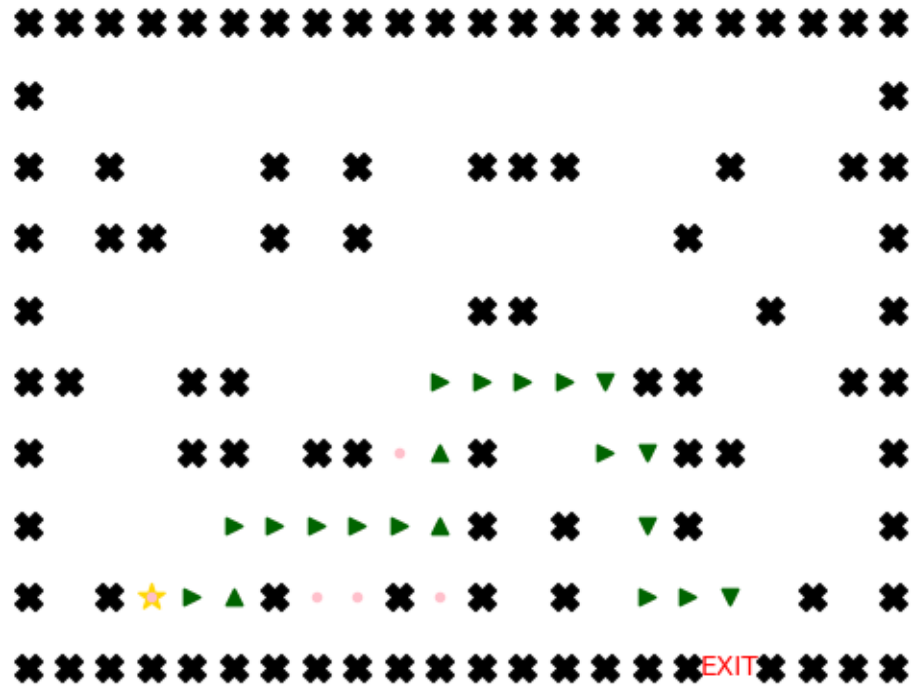
Tổng chi phí: 123

Thuật toán BFS

Độ dài đường đi: 22

Tổng chi phí: 104

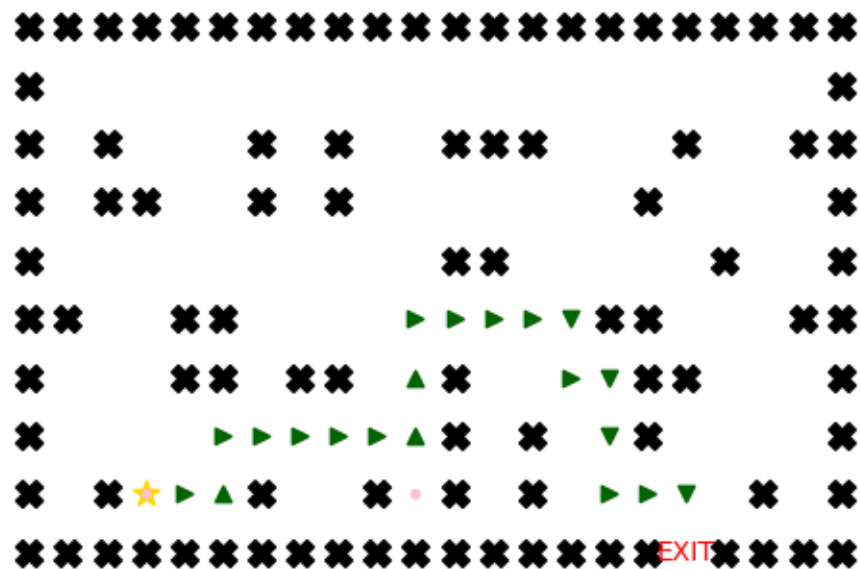
Thuật toán GBFS Manhattan



Độ dài đường đi: 22

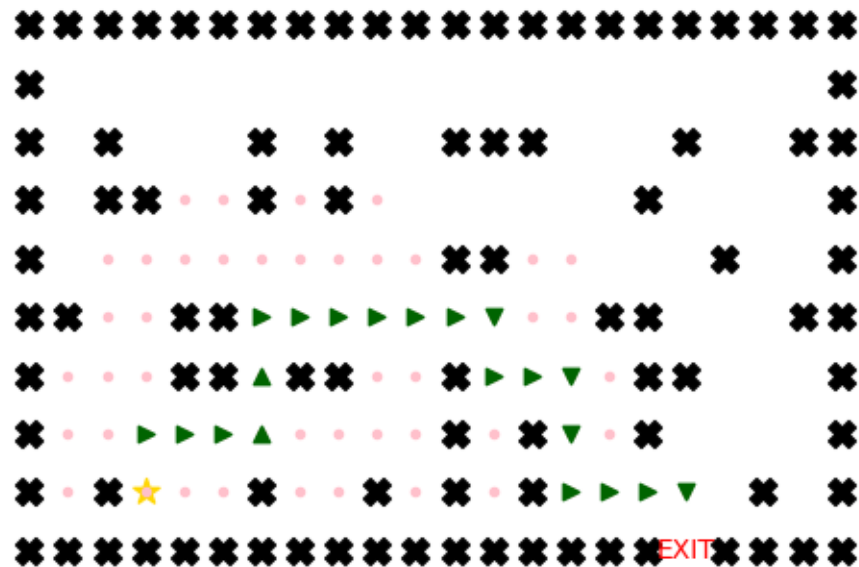
Tổng chi phí: 25

Thuật toán GBFS Euclid



Độ dài đường đi: 22

Tổng chi phí: 22

Thuật toán A* Euclid

Độ dài đường đi: 22

Tổng chi phí: 61

Nhận xét

- Kết quả của bản đồ 2 cũng không có nhiều sự sai khác so với bản đồ 1, thuật toán GBFS cũng cho ta kết quả tốt, đặc biệt là GBFS Euclid với tổng chi phí bằng độ dài đường đi.
- BFS và A* Euclid đều bảo đảm được độ dài đường đi nhưng chi phí di chuyển vẫn tốn khá nhiều.
- Thuật toán DFS vẫn cho thấy sự không hiệu quả khi đã duyệt toàn bộ các điểm có thể nhưng vẫn cho ra đường đi rất dài.

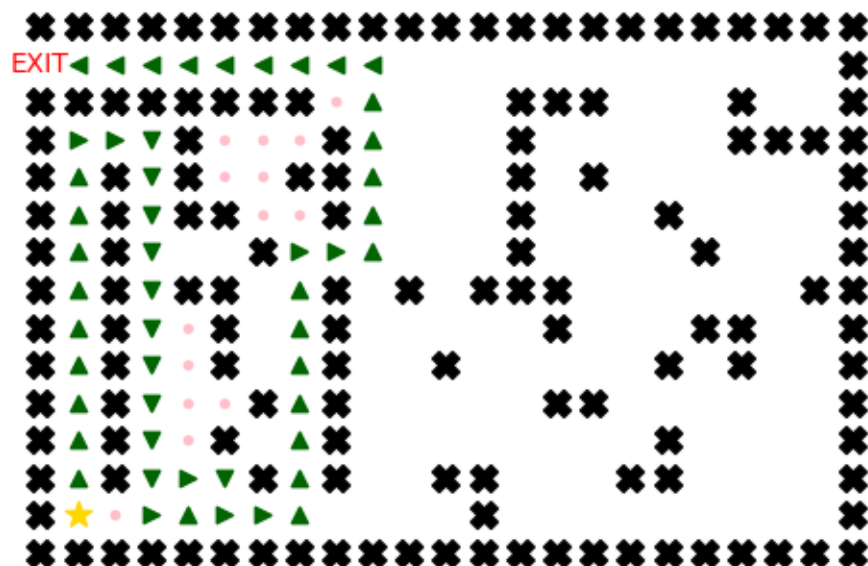
Bản đồ 3



Điểm bắt đầu $(x, y) = (13, 1)$

Điểm kết thúc $(x, y) = (1, 0)$

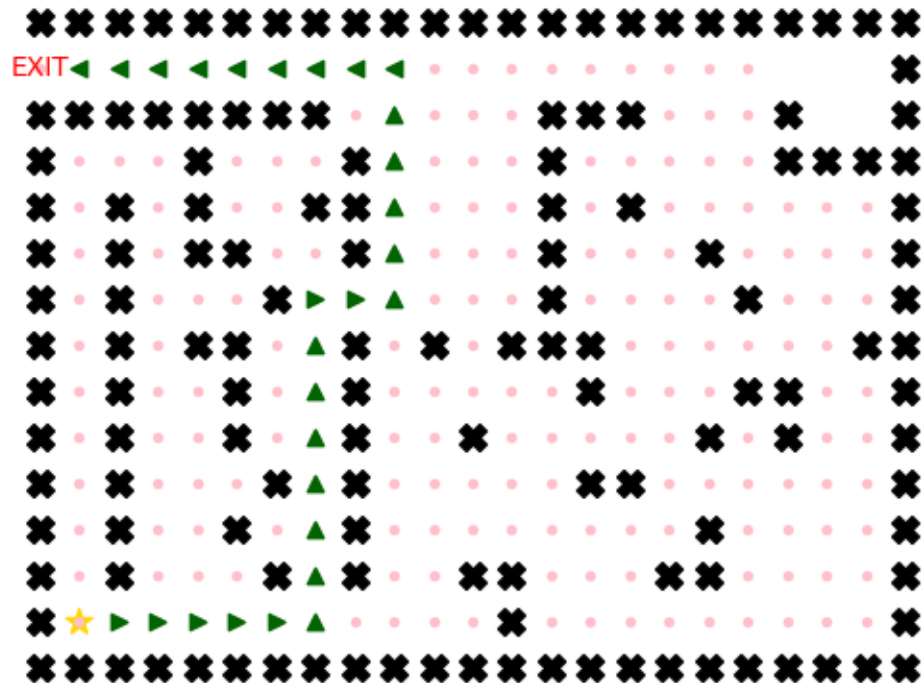
Thuật toán DFS



Độ dài đường đi: 52

Tổng chi phí: 66

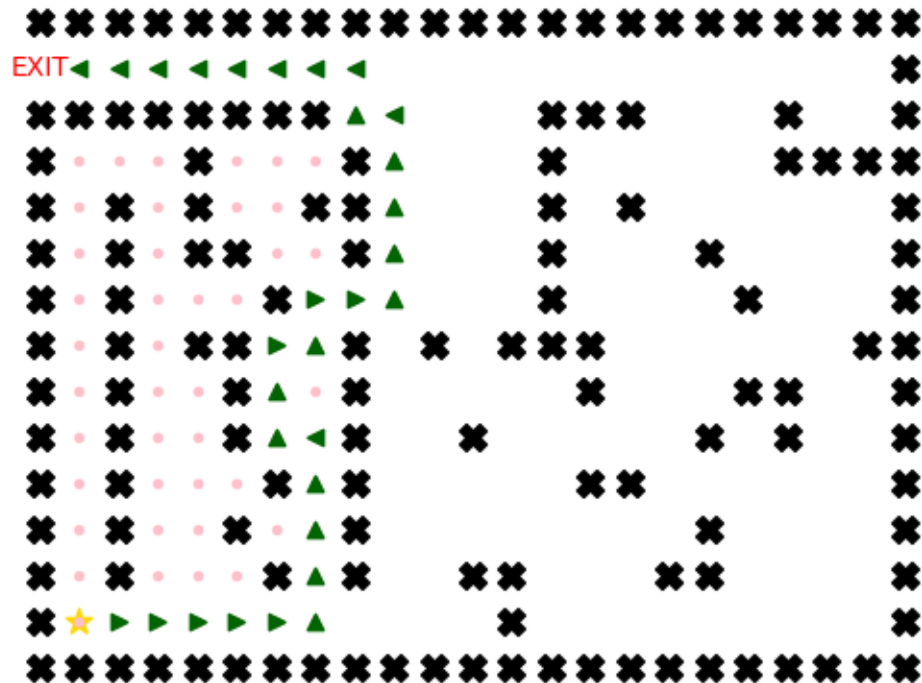
Thuật toán BFS



Độ dài đường đi: 30

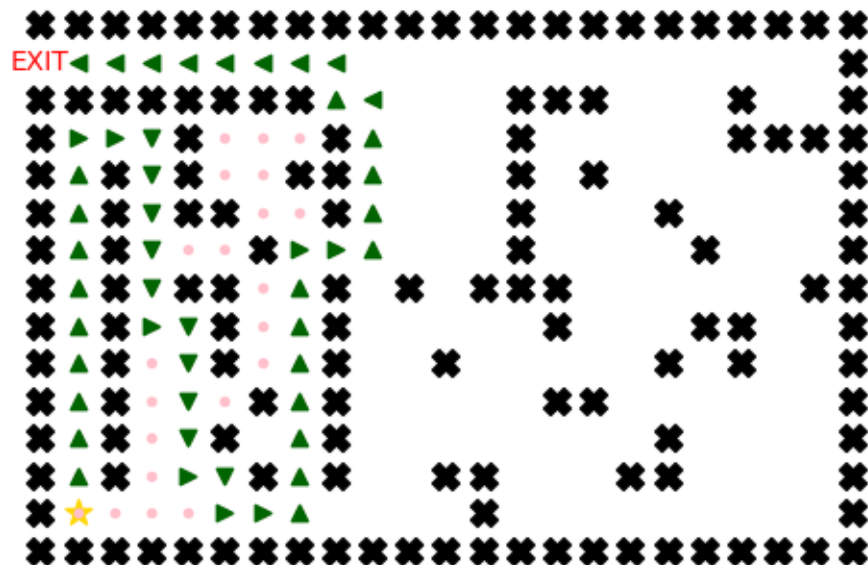
Tổng chi phí: 198

Thuật toán GBFS Manhattan



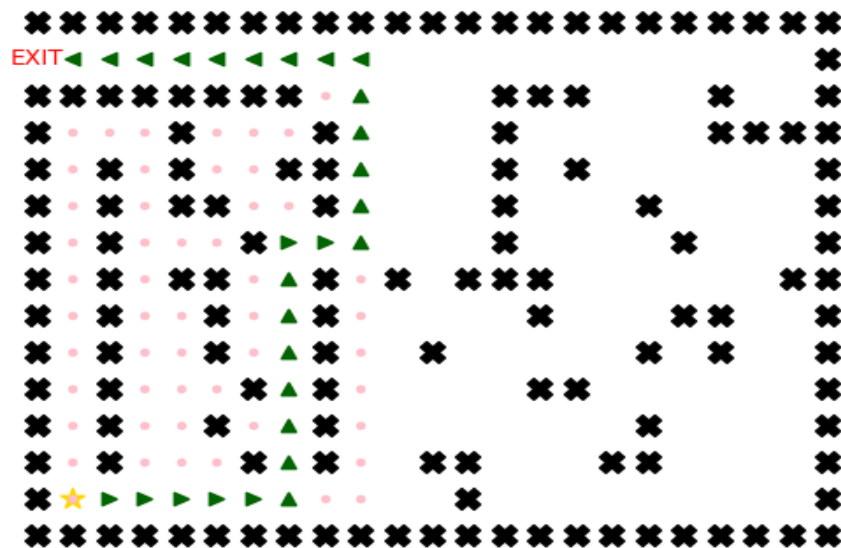
Độ dài đường đi: 32

Tổng chi phí: 70

Thuật toán GBFS Euclid

Độ dài đường đi: 50

Tổng chi phí: 69

Thuật toán A* Manhattan

Độ dài đường đi: 30

Tổng chi phí: 79

Thuật toán A* Euclid

Độ dài đường đi: 30

Tổng chi phí: 95

Nhận xét

- Thuật toán DFS cho ta chi phí thấp nhất nhưng đường đi vẫn chưa tối ưu.
- Thuật toán A* và BFS cho ta đường đi tối ưu với độ dài ít hơn là 30, nhưng vẫn bỏ ra nhiều chi phí để duyệt các điểm phía bên phải bản đồ.
- Trong trường hợp này, A* Manhattan hoạt động tốt hơn so với A* Euclid về mặt chi phí. Có thể nói A* Manhattan cho ta một kết quả tốt.
- GBFS Manhattan cho ta đường đi khá tốt (32) và chi phí ổn (70).
- Kết quả của GBFS Euclid là một kết quả trung bình với đường đi chỉ tốt hơn DFS 1 chút còn chi phí chỉ nhiều hơn DFS 1 chút.

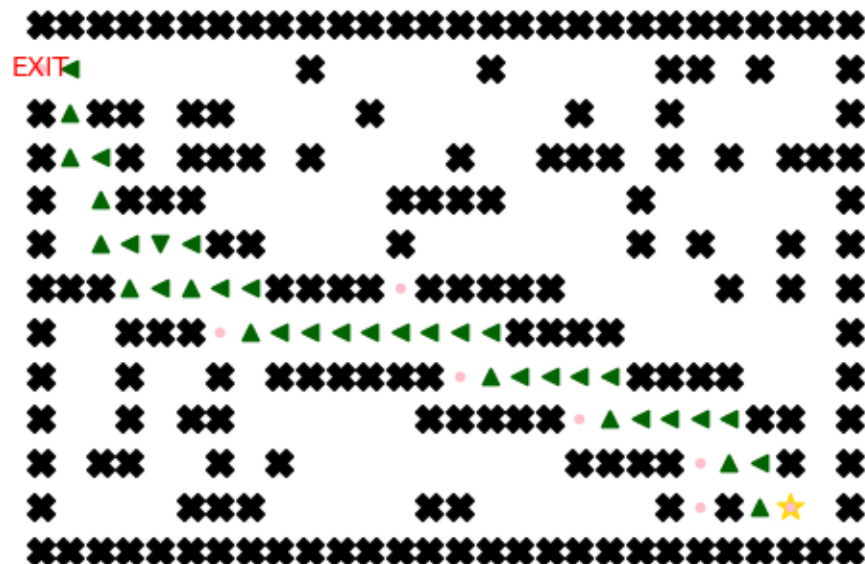
Bản đồ 4



Điểm bắt đầu $(x, y) = (11, 25)$

Điểm kết thúc $(x, y) = (1, 0)$

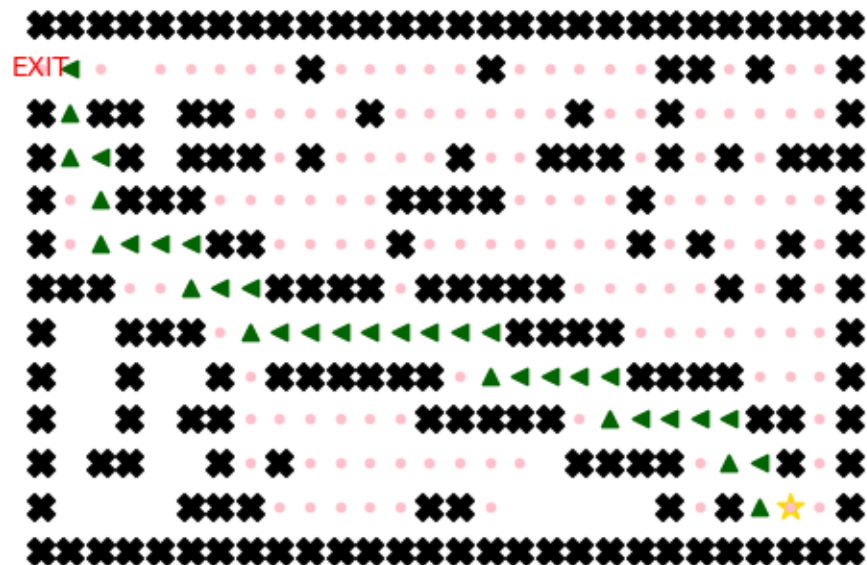
Thuật toán DFS



Độ dài đường đi: 38

Tổng chi phí: 44

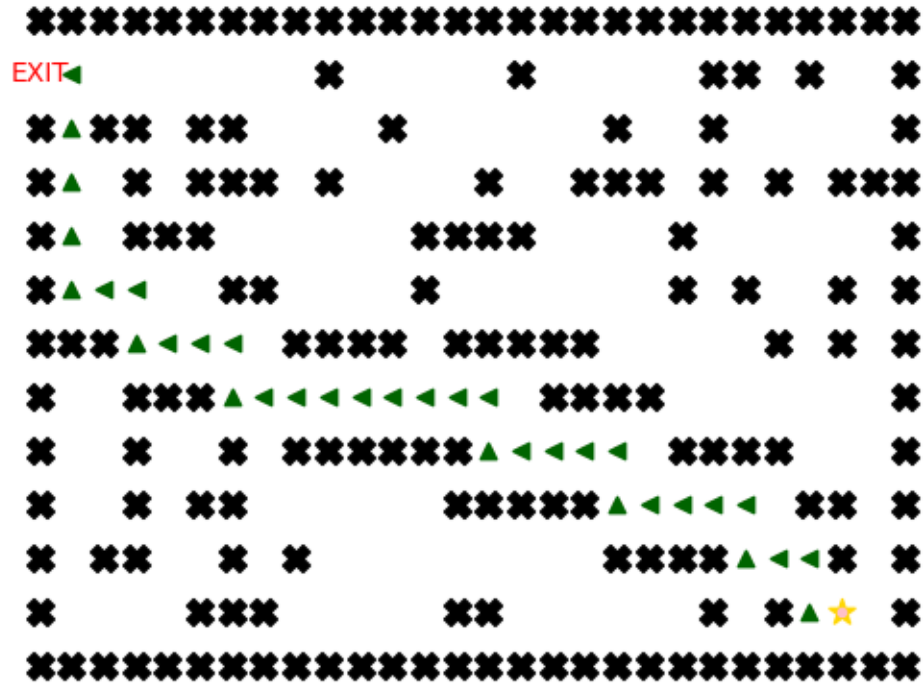
Thuật toán BFS



Độ dài đường đi: 36

Tổng chi phí: 165

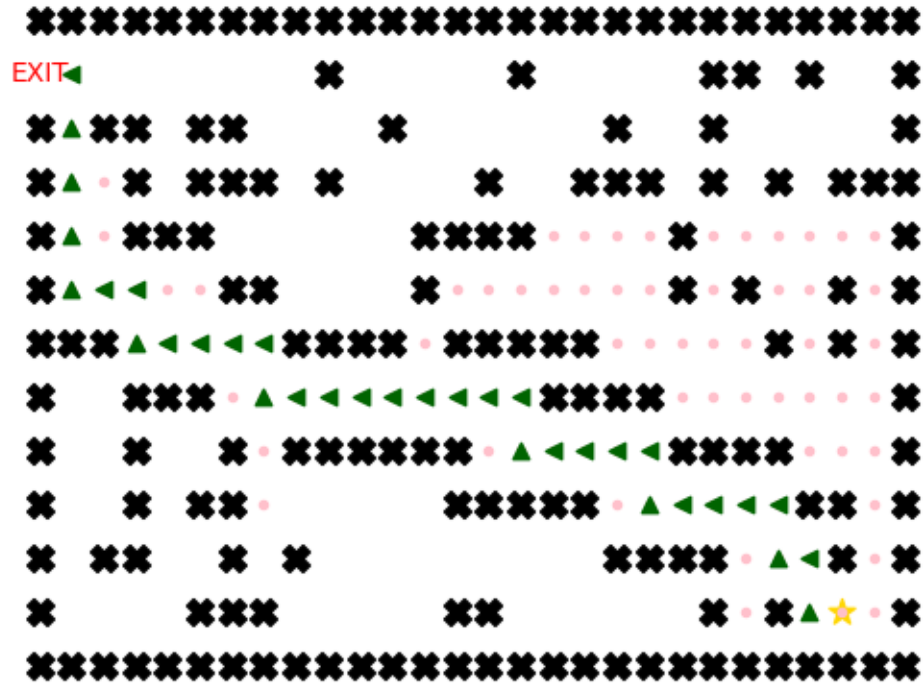
Thuật toán GBFS Manhattan



Độ dài đường đi: 36

Tổng chi phí: 35

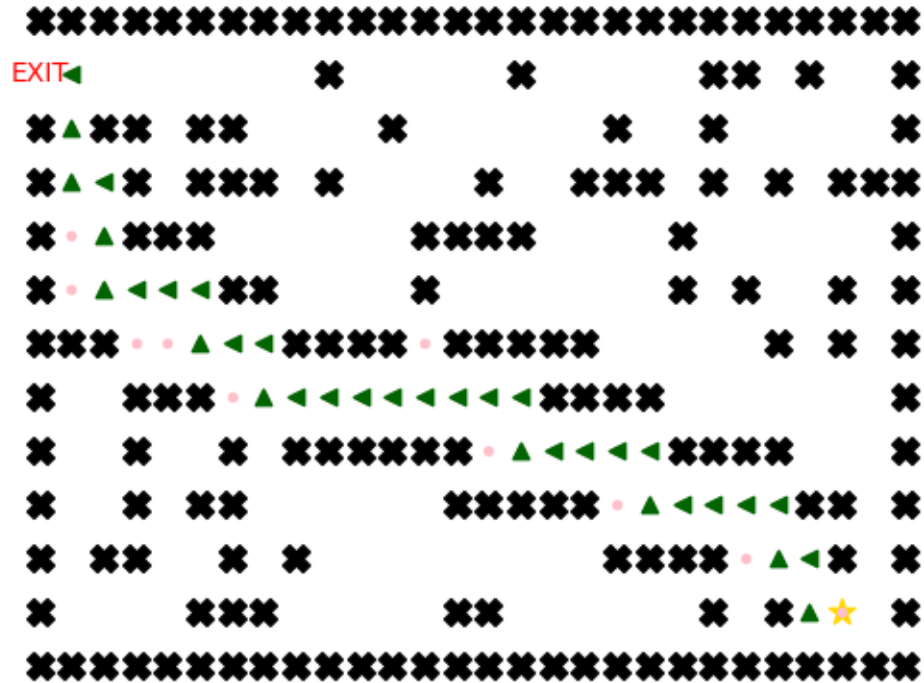
Thuật toán GBFS Euclid



Độ dài đường đi: 36

Tổng chi phí: 88

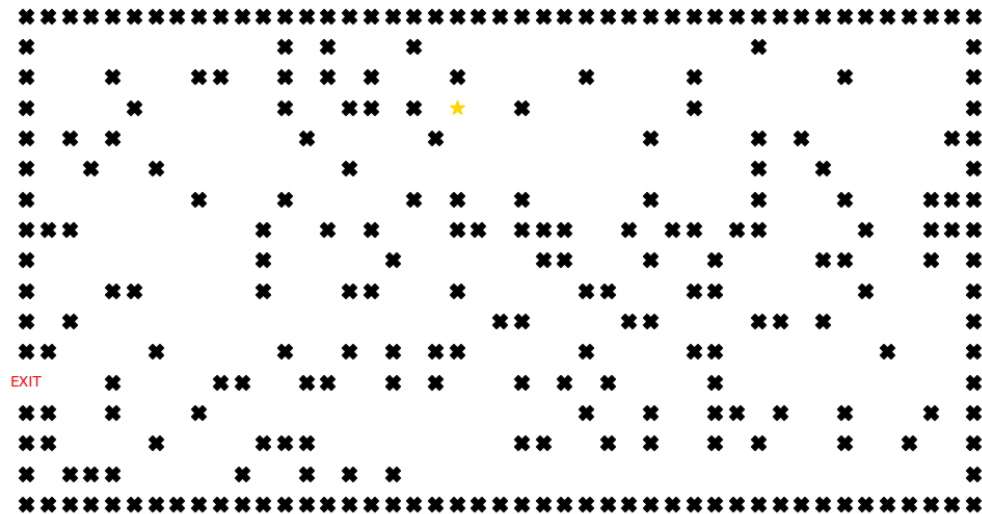
Thuật toán A* Manhattan



Độ dài đường đi: 36

Tổng chi phí: 44

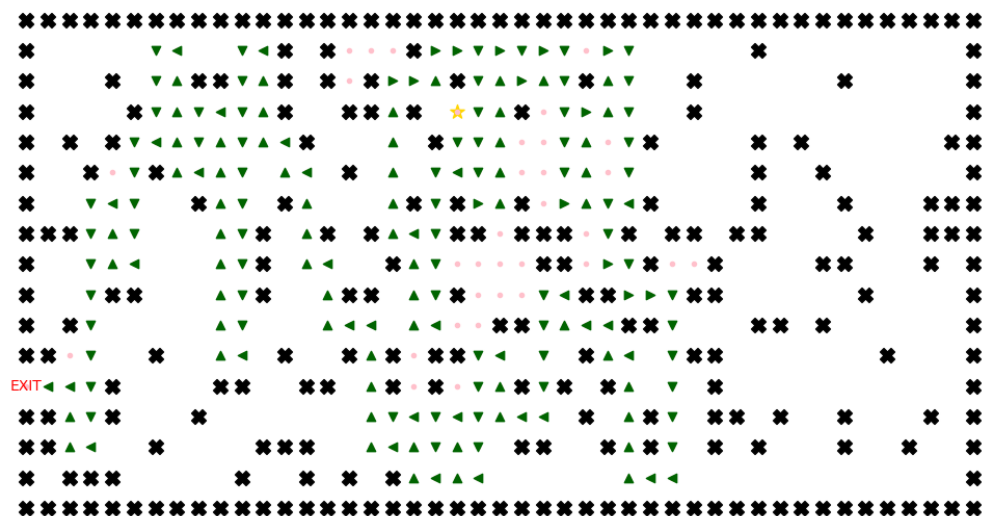
Bản đồ 5



Điểm bắt đầu $(x, y) = (3, 20)$

Điểm kết thúc $(x, y) = (12, 0)$

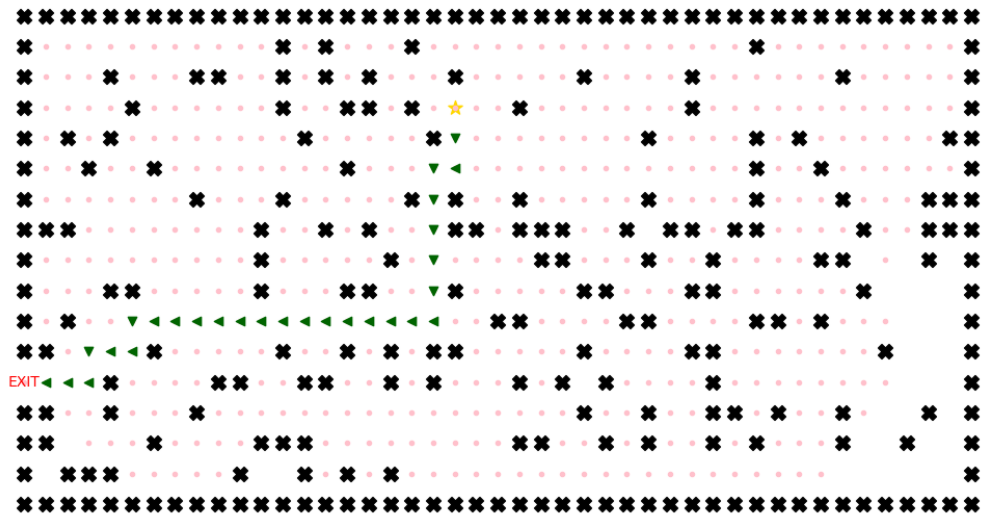
Thuật toán DFS



Độ dài đường đi: 180

Tổng chi phí: 212

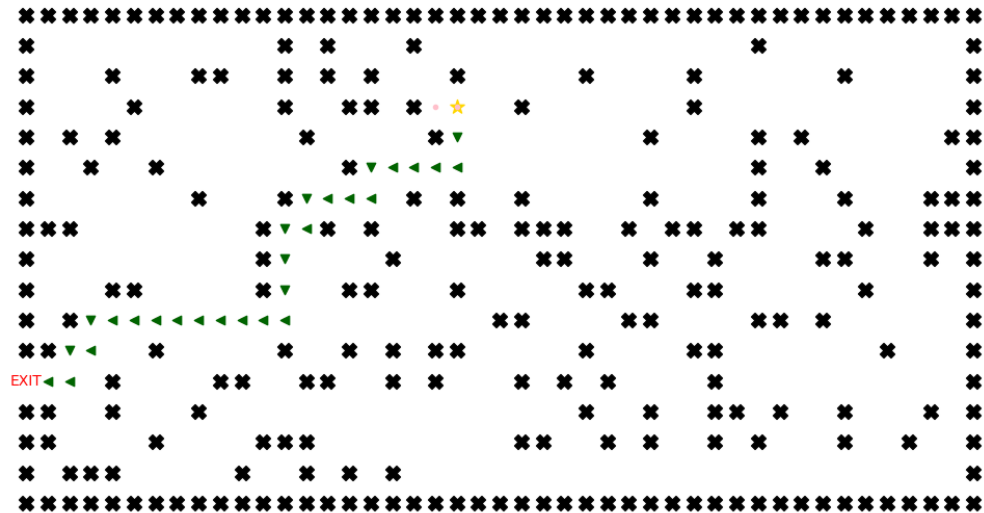
Thuật toán BFS



Độ dài đường đi: 30

Tổng chi phí: 469

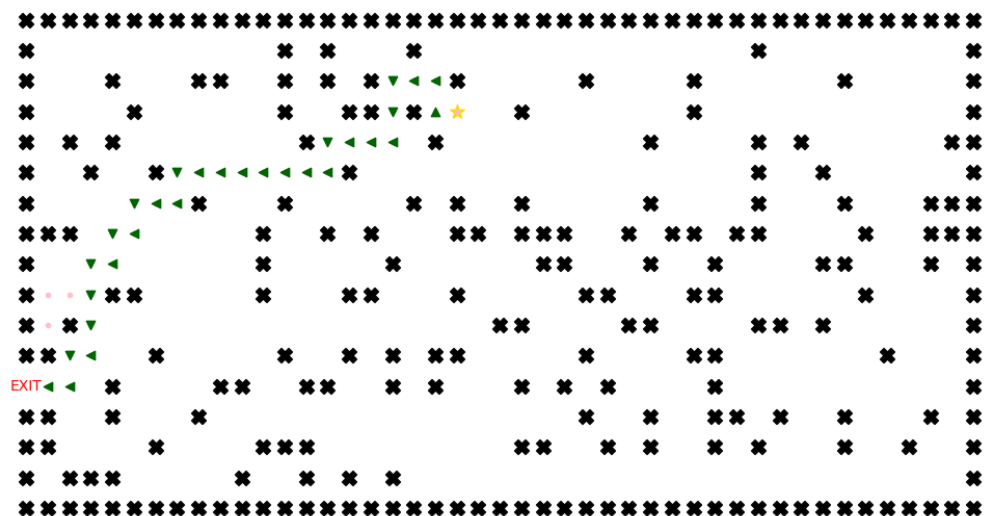
Thuật toán GBFS Manhattan



Độ dài đường đi: 30

Tổng chi phí: 30

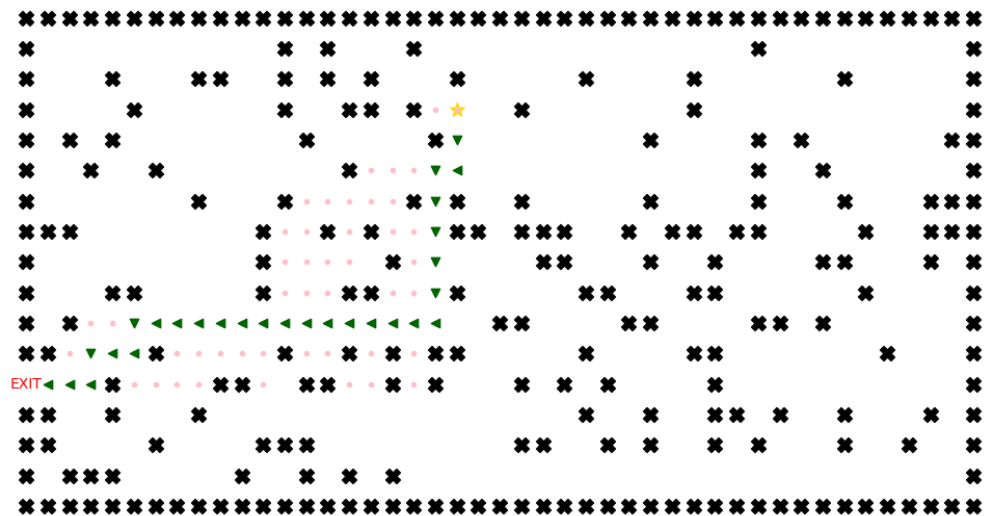
Thuật toán GBFS Euclid



Độ dài đường đi: 32

Tổng chi phí: 34

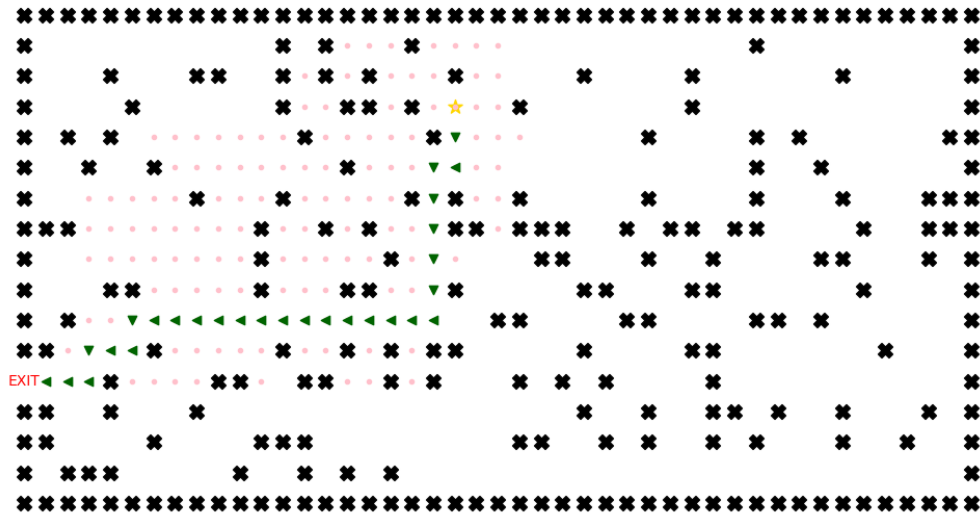
Thuật toán A* Manhattan



Độ dài đường đi: 30

Tổng chi phí: 73

Thuật toán A* Euclid



Độ dài đường đi: 30

Tổng chi phí: 151

Nhận xét

- Thuật toán GBFS Manhattan cho ta kết quả hoàn hảo nhất có thể với số bước đi và chi phí đều bằng 30.
- Thuật toán GBFS Euclid cũng cho ta một kết quả tốt với đường đi gần như tối ưu (độ dài đường đi: 32) và chi phí cũng khá tốt (34).
- Thuật toán A* cho ta kết quả là đường đi có độ dài thấp nhất (30 như GBFS Manhattan) nhưng chi phí cao hơn (73 với Manhattan và 151 với Euclid)
- BFS trong trường hợp này cũng thu về được đường đi tối ưu (độ dài: 30) nhưng vẫn tốn rất nhiều chi phí (469, cao nhất) khi mà chúng duyệt một phần bản đồ rất lớn so với A* và GBFS.
- DFS cho ta đường đi tệ nhất khi lựa chọn hướng đi khác với các thuật toán kia dẫn tới việc kết quả không tối ưu (độ dài đường đi: 180, cao nhất trong các thuật toán) nhưng chi phí khá tốt so với đường đi (212)

3 Nhận xét chung trên bản đồ không điểm thưởng

Đánh giá trên từng thuật toán

- **Thuật toán DFS:** thường cho kết quả đường đi có chi phí lớn hơn so với các thuật toán còn lại. Vì là duyệt theo chiều sâu, khi chọn 1 hướng để đi tiếp thì nó sẽ duyệt cho đến khi không có đường đi tiếp theo hoặc khi đến đích nên rất khó để dự đoán kết quả. Số lượng điểm phải duyệt rất lớn. Đường đi tối ưu khi dùng thuật toán này phụ thuộc vào hướng mở rộng được lựa chọn trước của điểm đang xét.
- **Thuật toán BFS:** có xu hướng mở rộng đường đi ra các phía xung quanh (4 phía) đều nhau (theo 1 bước, 2 bước, ...), số điểm duyệt qua là lớn nhất. Trong trường hợp chi phí đường đi tại mỗi điểm là như nhau thì đây là thuật toán tối ưu nhất.
- **Thuật toán GBFS:** là thuật toán luôn chọn đường đi xuất hiện tốt nhất mỗi thời điểm, là sự kết hợp của DFS và BFS, sử dụng một hàm heuristic để đánh giá chi phí. Hàm heuristic được chọn có ảnh hưởng lớn đến việc tìm đường đi tối ưu, khi nó có xu hướng đi tới điểm có chi phí ước lượng gần đích nhất.
- **Thuật toán A*:** có xu hướng mở rộng điểm hướng về đích, đối với thuật toán tìm kiếm có thông tin thì số điểm duyệt là nhiều hơn so với GBFS (để đảm bảo số bước đi là ít nhất). Nếu chọn heuristic hợp lý thì nó đảm bảo tối ưu. Trong trường hợp chi phí mỗi điểm như nhau và bằng 1 thì 2 hàm heuristic là Euclide và Manhattan sẽ đảm bảo việc A* sẽ cho đường đi tối ưu nhất. Điểm hạn chế là tốn nhiều bộ nhớ để lưu trạng thái điểm (như các chiến lược tìm kiếm theo chiều rộng)

Đánh giá 2 hàm heuristic

- **Manhattan**: số điểm được duyệt ít hơn so với Euclid, các khoảng cách tịnh tiến dx , dy phần lớn là các bước di chuyển lên, phải, xuống, trái. Nếu có nhiều chướng ngại vật thì chi phí sẽ lớn hơn tổng dx và dy .
- **Euclid**: số điểm được duyệt nhiều hơn, chi phí luôn lớn hơn hoặc bằng chi phí ước tính (trong trường hợp đường thẳng nối điểm bất kỳ và đích song song với 1 trục) và luôn lớn hơn chi phí ước tính (trong trường hợp đường thẳng nối điểm bất kỳ và đích cắt các trục).

4

Tài liệu tham khảo

- Bộ bài giảng môn Cơ sở Trí tuệ nhân tạo, Trường Đại học Khoa học Tự nhiên, ĐHQG-HCM
- Python 3 documentation: <https://docs.python.org/3/>
- Artificial Intelligence: a modern approach, 4th, Stuart J.Russell and Peter Norvig.