



KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN MÔN HỌC
CƠ SỞ TRÍ TUỆ NHÂN TẠO

THUẬT TOÁN TÌM KIẾM

Trợ giảng:

Nguyễn Khánh Toàn
ktoan271199@gmail.com
Lê Minh Nhật
minhnhatvt2@gmail.com

Nội dung của bài tập

- Nghiên cứu, cài đặt và trình bày các thuật toán tìm kiếm đường đi.

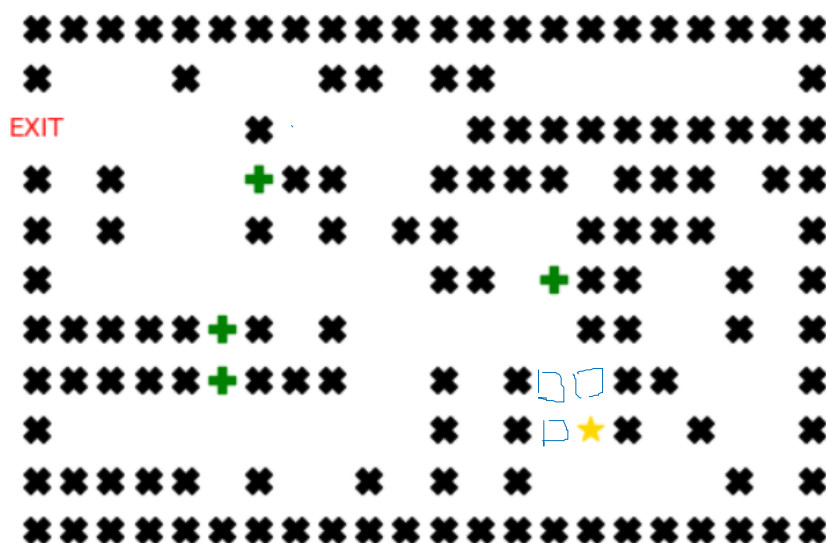
Quy định

- Tối đa 2 thành viên/nhóm.
- Thời gian: 3 tuần (xem chi tiết trên Moodle).
- Thư mục bài làm: nếu nhóm có 1 sinh viên thì đặt tên là <MSSV>, nếu 2 sinh viên thì đặt tên là <MSSV1>_<MSSV2>, bao gồm các nội dung sau:
 - Báo cáo trình bày các vấn đề được yêu cầu trong đề bài, định **dạng pdf**. Trang đầu tiên ghi thông tin nhóm (tên thành viên, mã số sinh viên).
 - Mã nguồn chương trình cài đặt và hướng dẫn chạy: đặt trong thư mục Source, bao gồm các file mã nguồn sử dụng để làm bài tập. **Ngôn ngữ: Python 3**.
 - Thư mục kiểm thử: Chứa những bản đồ mà các bạn đã tự thiết kế (là file văn bản định dạng **.txt** sẽ được trình bày dưới đây).
- Nén thư mục bài làm thành định dạng zip rồi nộp trên Moodle.

Yêu cầu chi tiết

Cho một bản đồ mê cung như hình vẽ [1](#), trong đó:

- Ký hiệu **ngôi sao** đại diện cho vị trí xuất phát của tác nhân,
- Ký hiệu **x** thể hiện các bức tường và tác nhân sẽ không thể di chuyển lên các vị trí này.
- Ký hiệu **+** thể hiện các ô điểm thưởng mà nếu tác nhân di chuyển vào các ô này sẽ được **giảm chi phí thực hiện đường đi**.



Hình 1: Bản đồ trò chơi.

- Ký hiệu **exit** đại diện cho vị trí đích mà tác nhân cần di chuyển đến, là vị trí **duy nhất** mà tác nhân có thể thoát khỏi mê cung.

Mục tiêu của các bạn sinh viên là cài đặt các thuật toán tìm kiếm đường đi từ vị trí xuất phát đến vị trí đích (vị trí thoát khỏi mê cung) cho tác nhân. Trong đó tác nhân chỉ có thể di chuyển lên, xuống, trái, phải với **chi phí bằng nhau**. Các bạn cần cài đặt các thuật toán sau:

- Thuật toán tìm kiếm không có thông tin:
 - Thuật toán tìm kiếm DFS (Depth First Search).
 - Thuật toán tìm kiếm BFS (Breadth First Search).
- Thuật toán tìm kiếm có thông tin:
 - Thuật toán tìm kiếm tham lam (Greedy Best First Search).
 - Thuật toán tìm kiếm A*.

Đối với các thuật toán tìm kiếm có thông tin, các bạn **tự định nghĩa hàm heuristic** phù hợp cho bài toán này. Các bạn có thể **thử nhiều hàm heuristic khác nhau** và báo cáo kết quả.

Về bản đồ trò chơi, sẽ có hai loại: bản đồ không có điểm thưởng và bản đồ có điểm thưởng. Nhiệm vụ của các bạn là cài đặt các thuật toán tìm kiếm và tự thiết kế các bản đồ cho hai loại trên.

- Với trường hợp bản đồ **không có điểm thưởng** (80%), các bạn sẽ cài đặt 4 thuật toán trên để giải quyết. Các bạn sẽ tự thiết kế và báo cáo khoảng 5 bản đồ tiêu biểu (các bạn nên chọn những bản đồ mà các thuật toán có sự khác biệt với nhau nhiều) và **số sánh** sự khác nhau giữa cách tìm đường đi trong các loại bản đồ này với từng chiến lược khác nhau. Cụ thể với một bản đồ được chọn để báo cáo, các bạn sẽ nhận xét:
 - Sự **khác nhau giữa các chiến lược** tìm kiếm đối với bản đồ này được thể hiện như thế nào? Các bạn cần có hình vẽ **bản đồ output đường đi** để minh họa và vận dụng lý thuyết về **tính đầy đủ và tối ưu** của các thuật toán để giải thích.
 - Với các chiến lược tìm kiếm có thông tin, các bạn có thể **chọn nhiều hàm heuristic** khác nhau và báo cáo sự khác nhau giữa các chiến lược đối với từng hàm heuristic.
- Với trường hợp bản đồ có điểm thưởng (20%), các bạn sẽ suy nghĩ cách giải quyết và đề xuất chiến lược để tác nhân di chuyển sao cho chi phí đường đi từ điểm bắt đầu đến điểm thoát khỏi mê cung là **nhỏ nhất** có thể (lưu ý rằng tác nhân không nhất thiết phải đi qua hết các điểm thưởng). Các bạn cần thiết kế bản đồ với tối thiểu 3 trường hợp: có 2, 5, 10 điểm thưởng với giá trị khác nhau trên bản đồ. Nếu các bạn **không thể** tìm được lời giải tối ưu (bất kể trong trường hợp số lượng điểm thưởng là nhiều hay ít), hãy đề xuất chiến lược heuristic để giải quyết, chẳng hạn **tham lam ăn tất cả điểm thưởng theo độ lớn giá trị** của chúng rồi mới tìm đường thoát khỏi mê cung.
- Kích thước của các bản đồ được chọn để báo cáo phải đảm bảo có ít nhất một bản đồ lớn (chiều dài và chiều rộng của bản đồ phải có một số ít nhất bằng 15 và một số ít nhất bằng 35).

- Số lượng bản đồ tối thiểu cần báo cáo và phân tích: 5 bản đồ đối với bản đồ không có điểm thưởng và 3 bản đồ đối với bản đồ có điểm thưởng.

Các bạn tham khảo hàm vẽ bản đồ ở notebook sau: [link](#)

Thiết kế kịch bản kiểm thử và bản đồ

Mỗi kịch bản kiểm thử được thiết kế như sau:

- Dòng đầu là số lượng điểm thưởng n ($n = 0$ với bản đồ không có điểm thưởng).
- n dòng tiếp theo, mỗi dòng sẽ bao gồm 3 số nguyên x, y, z với x, y là tọa độ của điểm thưởng trong ma trận; z là giá trị của điểm thưởng, sẽ là các số nguyên âm.
- Các dòng tiếp theo mô tả bản đồ của trò chơi. Các bạn lưu ý điểm kết thúc của hành trình sẽ là điểm thoát khỏi mê cung (ví dụ trong hình 2 thì điểm kết thúc sẽ là điểm ở dòng 2 và cột 0); điểm bắt đầu của tác nhân được ký hiệu bằng ký tự **S**; các ký tự x sẽ là các bức tường; các ký tự $+$ sẽ là các điểm thưởng.

Ví dụ với file '**maze.txt**' như sau:

Lưu ý về điểm thoát khỏi mê cung: Các bạn lưu ý bản đồ mê cung phải được thiết kế sao cho tồn tại **duy nhất** một vị trí nằm trên **các cạnh của bản đồ** để tác nhân có thể thoát ra ngoài được.

Đầu ra cho mỗi bản đồ: Kết quả trả về cho mỗi bản đồ sẽ là đồ họa biểu diễn **đường đi** và **chi phí thực hiện** đường đi. Ví dụ về hình vẽ đường đi kết quả được biểu diễn ở hình 3.

Đánh giá

- Điểm đồ án được tính như sau:
 - **Bản đồ không có điểm thưởng (8 điểm):** Các bạn hoàn thành cài đặt 4 thuật toán tìm đường đi, tự thiết kế bản đồ để báo cáo và nhận xét.

```

maze.txt - Notepad
File Edit Format View Help
4
3 6 -3
5 14 -1
6 5 -2
7 5 -4
XXXXXXXXXXXXXXXXXXXXX
X  X  XX XX      X
      X      XXXXXXXXX
X X  +XX  XXXX XXX XX
X X  X X XX  XXXX  X
X      XX +XX  X X
XXXXX+X X      XX  X X
XXXXX+XXX  X X  XX  X
X      X X Sx X  X
XXXXX  X  X X X      X X
XXXXXXXXXXXXXXXXXXXXX

```

Hình 2: Minh họa về tập tin đầu vào của mỗi bản đồ.



Hình 3: Ví dụ về output đường đi thoát khỏi mê cung.

- **Bản đồ có điểm thưởng (2 điểm):** Các bạn đề xuất thuật toán, cài đặt, tự thiết kế bản đồ để báo cáo và nhận xét.
- Báo cáo cần trình bày rõ ràng, mạch lạc; mã nguồn cần được tổ chức rõ ràng.
- **Điểm cộng (1 điểm):** Các bạn suy nghĩ kịch bản khó hơn cho trò chơi, ví dụ như bản đồ sẽ có các cánh cửa để dịch chuyển bất kỳ từ điểm (i, j) sang điểm (i', j') , hoặc các điểm chướng ngại vật trên bản đồ có thể di chuyển. Với kịch bản mà các bạn nghĩ ra, các bạn tìm lời giải tối ưu nhất có thể cho kịch bản đó, sau đó tự thiết kế bản đồ (tối thiểu 3 bản đồ) và nhận xét.
- Tổng điểm tối đa là 11/10 (1 điểm cộng). Nhóm nào quá 10 điểm sẽ được tính là 10 điểm.
- Báo cáo/mã nguồn có tham khảo cần phải ghi nguồn rõ ràng ở cuối báo cáo.
- Các bài giống nhau sẽ bị 0 điểm môn học.

Liên hệ

Mọi thắc mắc trong quá trình thực hiện vui lòng gửi mail về ktoan271199@gmail.com hoặc minhnhatvt2@gmail.com.