

# ĐỒ ÁN 1 – TÌM KIẾM

---

## A. Quy định thực hiện đồ án

- Nội dung đồ án về **chiến lược tìm kiếm mù** (uninformed search) và **tìm kiếm heuristic** (informed search). Đồ án có tính thử thách cao, đòi hỏi sự đầu tư của người thực hiện.
- Đồ án làm theo **CÁ NHÂN**.
- Thời gian thực hiện: **3 TUẦN**. Xem ngày nộp cụ thể tại link Moodle
- Thang điểm: 100 + 20 điểm (để giáo viên dễ chấm, không có gì đặc biệt đâu!!!)
- Nội dung cần nộp:
  - Báo cáo đồ án dành cho Yêu cầu 1:
    - Tên: Report\_1\_1. Định dạng: .doc/.docx/.pdf
    - Trang đầu tiên ghi rõ thông tin cá nhân (họ tên, MSSV) và các milestone thực hiện đồ án.
    - Mô tả thiết kế lưu trữ đồ thị và các thông tin liên quan.
    - Cung cấp 1-2 ví dụ đơn giản cho mỗi thuật toán để minh họa input và output của chương trình đã cài đặt.
    - Các lưu ý đặc biệt để chạy chương trình thành công
  - Báo cáo đồ án dành cho Yêu cầu 2:
    - Tên: Report\_1\_2. Định dạng: .doc/.docx/.pdf
    - Trang đầu tiên ghi rõ thông tin cá nhân (họ tên, MSSV) và các milestone thực hiện đồ án.
    - Mô tả thiết kế lưu trữ bản đồ và các thông tin liên quan.
    - Mô tả thuật toán được sử dụng: ý tưởng, mã giả, chạy 1-2 ví dụ đơn giản để minh họa, và xét các trường hợp đặc biệt (nếu có).
    - Mô tả chức năng các hàm được cài đặt trong mã nguồn.
    - Hướng dẫn chi tiết để chạy chương trình thành công. Trình bày ít nhất 3 ví dụ cho trường hợp tìm đường đi thành công và thất bại (nếu có).
  - Mã nguồn của Yêu cầu 1 và Yêu cầu 2
    - Mã nguồn của mỗi yêu cầu được tổ chức thành một solution riêng.
    - Giữ cấu trúc toàn bộ thư mục của project, xóa các file trung gian tạo ra trong quá trình build project.
    - Xóa tập tin .EXE vì file này rất dễ nhiễm virus.
  - Báo cáo và mã nguồn đặt trong thư mục đặt tên <X>\_1 với X là MSSV (ví dụ 1502001\_1) và 1 là Đồ án 1 (cố định). Nén thư mục về định dạng .rar hoặc .zip.
- Lưu ý:

- Các trường hợp sao chép dù chỉ là một phần trong project đều có thể khiến bài làm bị 0 điểm và có thể nhận hình thức kỷ luật nghiêm trọng hơn tùy mức độ.
- Sinh viên tự kiểm tra bài làm trước khi nộp. Giáo viên không giải quyết sai sót về phiên bản của bài làm.
- Tuân thủ nghiêm ngặt quy định về nội dung cần nộp (dữ liệu nộp, quy ước đặt tên,...) và mô tả yêu cầu vì sẽ có cột điểm dành cho vấn đề này.

## B. Yêu cầu 1: Triển khai các chiến lược tìm kiếm trên đồ thị (30 điểm)

Cho tập tin **input.txt** chứa thông tin về một đồ thị có hướng. Tập tin có định dạng như sau:

- Dòng đầu tiên chứa một số nguyên  $n$  ( $n \geq 3$ ) thể hiện số đỉnh của đồ thị.
- Dòng thứ hai chứa hai số nguyên biểu diễn chỉ mục của đỉnh nguồn và đỉnh đích, biết rằng chỉ mục của đỉnh nằm trong đoạn  $[0, n-1]$ .
- $n$  dòng tiếp theo chứa  $n$  số nguyên trên mỗi dòng để biểu diễn ma trận kề, các số nguyên cách nhau bằng khoảng trắng. Gọi  $[i, j]$  là giá trị tại dòng  $i$  cột  $j$  ( $i, j = 0, \dots, n-1$ ).  $[i, j] = 0$ : không có cạnh nối từ đỉnh  $i$  đến đỉnh  $j$ ,  $[i, j] = x > 0$ : có cạnh nối từ  $i$  đến  $j$  với trọng số  $x$ .
- Dòng cuối cùng chứa  $n$  nguyên không âm để biểu diễn giá trị heuristic tương ứng cho các đỉnh từ 0 đến  $n-1$ , các số nguyên cách nhau bằng khoảng trắng. Giá trị heuristic đã được đảm bảo điều kiện chấp nhận được.

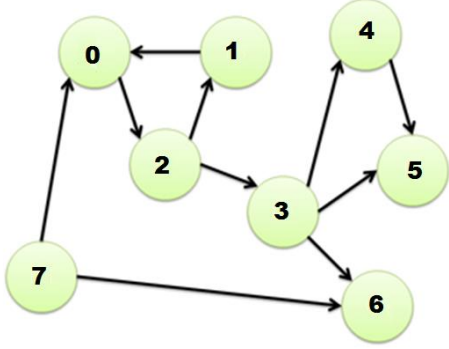
Lần lượt triển khai các chiến lược tìm kiếm như liệt kê bên dưới. Trong trường hợp nhiều đỉnh có **cơ hội chọn như nhau** thì **ưu tiên đỉnh có chỉ mục nhỏ hơn**. Ghi nhận lại thông tin về danh sách đỉnh đã duyệt và đường đi từ đỉnh nguồn đến đỉnh đích. Kết quả của mỗi chiến lược được lưu vào một tập tin riêng biệt.

- Tìm kiếm theo chiều sâu (depth-first search): có loại trừ lặp vô hạn (5 điểm)
- Tìm kiếm theo chiều rộng (breadth-first search) (5 điểm)
- Tìm kiếm chi phí đồng nhất (uniform-cost search) (5 điểm)
- Tìm kiếm tham lam (greedy best-first search) (5 điểm)
- Tìm kiếm  $A^*$  (10 điểm)

Tập tin đầu ra cho các chiến lược được đặt tên lần lượt là **dfs.txt**, **bfs.txt**, **ucs.txt**, **gbfs.txt** và **astar.txt**, đặt cùng cấp với tập tin \*.EXE, được phát sinh đồng thời khi chạy chương trình. Tập tin có định dạng như sau:

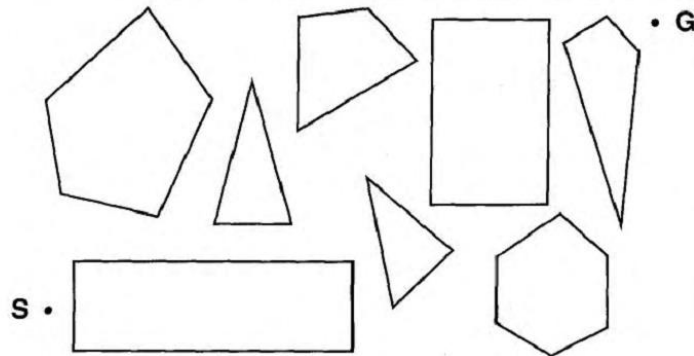
- Dòng đầu tiên chứa danh sách đỉnh đã duyệt, các đỉnh cách nhau bằng khoảng trắng. Đỉnh đã duyệt là đỉnh được lấy ra khỏi hàng đợi (ưu tiên) hoặc ngăn xếp.
- Dòng thứ hai chứa đường đi từ đỉnh nguồn đến đỉnh đích, các đỉnh cách nhau bằng khoảng trắng.

Ví dụ:

Đồ thị	input.txt	dfs.txt
	<pre> 8 1 6 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 5 6 4 2 8 7 0 2 </pre>	<pre> 1 0 2 3 4 5 6 1 0 2 3 6 </pre>

### C. Yêu cầu 2: Robot tìm đường (70 điểm)

Cho bản đồ phẳng xOy (góc phần tư I) trên hệ trục tọa độ 2 chiều. Người ta đặt trên bản đồ một điểm nguồn  $S(x_S, y_S)$  và một điểm đích  $G(x_G, y_G)$ , đồng thời đặt chướng ngại vật là các đa giác lồi sao cho các đa giác không được đặt chồng lên nhau hay có điểm chung.



Áp dụng chiến lược tìm kiếm (đã học hoặc tự tìm hiểu) để tìm đường đi ngắn nhất từ S đến G sao cho đường đi không được cắt xuyên qua các đa giác. Biết rằng:

- Robot có thể đi tùy ý theo mọi hướng, mỗi lần đi một đơn vị.
- Tọa độ của các đỉnh trên đa giác lồi được biết trước.
- Tọa độ có định dạng số nguyên dương.
- Bản đồ được giới hạn trong phạm vi từ tọa độ  $O(0, 0)$  đến  $(100,000; 100,000)$ .

Sinh viên được yêu cầu thực hiện các nội dung sau

- Thiết kế cấu trúc lưu trữ bản đồ (tức là tọa độ của các đa giác), điểm nguồn và đích. Tập tin lưu trữ cần ở định dạng mà người dùng khác có thể hiệu chỉnh được (ví dụ văn bản plain text .txt, .csv hoặc cơ sở dữ liệu .xls).
- Cài đặt thuật toán để tìm đường đi ngắn nhất, được chia thành các mức độ như sau
  - Level 1 (10 điểm): đọc bản đồ và trả lời có/không đường đi từ S đến G.
  - Level 2 (10 điểm): tính được độ dài đường đi theo khoảng cách Euclide trong không gian 2 chiều.

- Level 3 (20 điểm): biểu diễn đồ họa ở mức đơn giản nhất để người dùng có thể thấy được các đa giác và đường đi.
- Level 4 (20 điểm): đây là yêu cầu nâng cao không có tính bắt buộc. Các đa giác có thể di động với tốc độ h bước/s. Cách di động có thể ở mức đơn giản nhất là tới lui một khoảng nhỏ để đảm bảo không chồng lên đa giác khác (nhưng có thể đụng (2 đường/cạnh/đỉnh trùng nhau)). Đường đi sẽ thay đổi khi các đối tượng di chuyển (chiến lược thời gian thực). Giả sử sau khi đa giác di chuyển h tọa độ thì dừng lại 1s, trong thời gian đó hãy tính và vẽ lại đường đi.

Gợi ý thực hiện:

- Chương trình có thể nhận input dưới dạng file text bao gồm tọa độ đỉnh S, G và tập tọa độ các đỉnh đa giác.
- Xác định các đa giác nằm trên đường đi trực tiếp từ S đến G.
- Xem thêm gợi ý bài 3.15 trang 91 trong sách Artificial Intelligence: A Modern Approach.

## D. Tiêu chí đánh giá

- Báo cáo trình bày rõ ràng, đầy đủ về thông tin cá nhân, quá trình thực hiện và nội dung liên quan đến đề tài: **10 điểm**.
- Tuân thủ quy định nộp bài và trình bày đồ án: **10 điểm**
- Yêu cầu 1: **30 điểm**, chi tiết cho từng chiến lược đã đề cập ở mục trên
- Yêu cầu 2: Nêu được thuật toán sử dụng và giải thích được lí do sử dụng thuật toán. Khuyến khích thử nghiệm nhiều thuật toán và so sánh: **10 điểm**. Cài đặt được thuật toán để giải quyết bài toán: **40 điểm**. Tỷ lệ chi tiết theo từng level đã đề cập ở mục trên.
- Điểm cộng trong Yêu cầu 2: **20 điểm**