Thông tin cài đặt, chạy các chương trình demo

#by DTD for 2021 classes (128702 and 126523)

# I. Cài đặt

* Python 3.9 (trên máy tôi, python 3.10 có lỗi khi sử dụng thư viện matplotlib.pyplot)
  + Instal matplotlib
    - python -m pip install matplotlib
    - python -m pip install pycairo
* Pycharm: PyCharm 2021.2.2 (Community Edition)

# II. Blind search (blind\_search.py)

*Minh họa các phương pháp tìm kiếm mù (BFS, UCS, DFS, DLS, IDS), chỉ ra khác biệt trong duyệt các node trong quá trình tìm kiếm.*

## 1. Run option: search method = “BFS”

search\_method = "BFS" # BFS/UCS/DFS/DLS/IDS

Chart

Description automatically generated

* Duyệt node theo chiều sâu tăng dần (theo các hàng)
* Bộ nhớ lớn

## 2. Run option: search method = “UCL”

search\_method = "UCS" # BFS/UCS/DFS/DLS/IDS

Chart

Description automatically generated

* Duyệt các node theo chi phí đến node gốc tăng dần
* Chỉ số thực thi tương tự BFS

## 3. Run option: search method = “DFS”

search\_method = "DFS" # BFS/UCS/DFS/DLS/IDS

Chart

Description automatically generated

* Duyệt node sâu trước, theo các nhánh cây
* Bộ nhớ nhỏ

## 4. Run option: search method = “DLS” (limit = 4)

search\_method = "DLS" # BFS/UCS/DFS/DLS/IDS

Chart

Description automatically generated

* Duyệt node sâu trước đến độ sâu cho giới hạn (n=4)
* Bộ nhớ nhỏ

## 5. Run option: search method = “IDS”

search\_method = "IDS" # BFS/UCS/DFS/DLS/IDS

Chart

Description automatically generated

* Lặp lại DLS với giới hạn tăng dần
* Bộ nhớ nhỏ

# Run option: Tree\_cirle = True (chưa hoàn thiện)

Tree\_cirle = False # show tr

Chart, diagram

Description automatically generated

# III. Heuristic search (queencheck.py)

*Ví dụ về tìm kiếm với tri thức bổ sung (Informed search), thuật giải GA cho bài toán 8 quân hậu. Viết code theo thuật toán slide 42-43*

Có thể thuyết minh thêm về

* Thay đổi tính fitness: hàm fitness(board) (dòng 47) để quan sát tốc độ tìm ra lời giải (hiện đang có 2 option để thử)
* Chọn lựa cá thể (cho tái sx, lai ghép) theo fitness: theo phương pháp “roulette wheel selection” (hàm selectOne(population) dòng 61)

Diagram

Description automatically generated with medium confidence

Các options:

# demo option  
#1-show chromosome presentation and fitness calculation  
#2-show crossover operator  
#3-show mutation operator  
#4-run GA

## Option = 1 (show chromosome presentation and fitness calculation)

Text

Description automatically generated

## Option = 2 (show crossover operator)

Graphical user interface, text

Description automatically generated

## Option = 3 (show mutation operator)

A picture containing text, electronics, close

Description automatically generated

## Option = 4 (run GA)

Text

Description automatically generated

# IV. Tìm kiếm đối kháng (mininmax.py)

*Demo thuật toán minimax với trò chơi tic-tac-toe, phương thức hàm min/max, prunning. Các hàm min/max được viết (sát) theo slide lớp học*

Các options:

# demo option:  
# 0: show initial state  
# 1: show action and transition model  
# 1.5: terminal test and utility function: in the code  
# 2: run program

Size = 3  
Prunning = False  
DepthLimit = False  
Depth = 4

## Demo = 0 (Size = 3, Prunning = False): show initial state

Graphical user interface

Description automatically generated

## Demo = 1 (Size = 3, Prunning = False): show action and transition model

Text

Description automatically generated

## Demo = 2 (Size = 3, Prunning = False): run program

Text

Description automatically generated

## Other options and observations:

* Chạy minimax có prunning: prunning = True (quan sát thấy số move kiểm tra giảm đi)
* Chạy minimax với số ô lớn hơn (Size = 4, 5, có thể rất lâu)
* Sử dụng DepthLimit để tăng tốc độ

# V. Tìm kiếm đối kháng (chess.py)

*Demo thuật toán minimax với trò chơi cờ vua. Các hàm min/max được viết (sát) theo slide lớp học.*

*Giới thiệu cho sinh viên về ví dụ xây dựng trò chơi cờ vua (tham khảo để làm bài tập lớn): Có thể sử dụng thư viện về luật chơi (biểu diễn bàn cờ, quân cờ, danh sách nước đi hợp lệ); Tuyệt đối tự viết code tìm nước đi tối ưu (score, minimax, pruning...).*

Chương trình minh họa sử dụng thư viện (chess) của python để theo dõi, in trạng thái bàn cờ, liệt kê các nước đi hợp lệ. Các hàm tự viết để demo: min/max, tính điểm, ma trận vị trí quân cờ.

## Option = 0 (chess piece, board presentation, board display)

A screenshot of a computer

Description automatically generated with low confidence

## Option = 1 (chess piece, board presentation, board display)

Text

Description automatically generated

## Option = 3 (two agents playing)

Text

Description automatically generated

## Các trình bày khác

* Giới thiệu thư viện chess của python
* Hàm score tính điểm quân cờ (code)
* Ma trận tính điểm quân cờ theo vị trí (code)
* Giới thiệu về trò chơi cờ vua (dưới đây)
* Các nhóm (project) hiểu và phát triển các kỹ thuật nào?

# A picture containing diagram Description automatically generatedTrò chơi cờ vua

## Phát biểu bài toán

* Trạng thái bắt đầu
* Hàm chuyển trạng thái:
  + các nước đi hợp lệ
* Kiểm tra kết thúc trò chơi:
  + Chiếu hết? Hòa cờ? (thêm: hết giờ, phạm luật)
* Hàm tiện ích:
  + đánh giá thế cờ, giá trị quân cờ, vị trí quân cờ

Tham khảo: <https://en.wikipedia.org/wiki/Rules_of_chess>

## Ví dụ xây dựng hệ thống từ đầu (demo in python)

* Tự biểu diễn bàn cờ, quân cờ, nước đi
* Ví dụ hàm tìm nước đi hợp lệ, tính điểm bản cờ, thuật toán minimax
* Demo
  + # 0: chess piece, board presentation, board display
  + # 1: possible moves of pieces
  + # 2: score pieces and score board
  + # 3: play a move

## Ví dụ dùng thư viện (python chess library)

* Tham khảo: [https://python-chess.readthedocs.io/en/latest/#](https://python-chess.readthedocs.io/en/latest/)
* Ví dụ hàm tìm nước đi hợp lệ, tính điểm bản cờ, thuật toán minimax (như trên)
* Tính điểm theo vị trí quân cờ (ma trận điểm theo từng quân cờ, vị trí)

## Cân nhắc

* **Vấn đề đối với giới hạn độ sâu (ăn quân tại độ sâu tới hạn?)**
* **Thêm yếu tố ngẫu nhiên (vd chọn ngẫu nhiên trong các nước đi tốt nhất nếu gần score)**
* **Học từ CSDL các ván cờ mẫu**
  + **Link**
* **Quản lý thời gian tính toán (thời gian cho nước cờ, ván cờ)**
* Hiển thị GUI

## Lưu ý

* Có thể sử dụng thư viện về luật chơi (biểu diễn bàn cờ, quân cờ, danh sách nước đi hợp lệ)
* **Tuyệt đối tự viết code tìm nước đi tối ưu (score, minimax, pruning...)**