

## ĐỒ ÁN CUỐI KỲ

**Môn:** Nhập môn Trí tuệ Nhân tạo

**Thời gian làm bài:** 05 tuần

### I. Hình thức

- Đề tài giữa kỳ được thực hiện theo nhóm **04 – 05** sinh viên.
- Nhóm sinh viên thực hiện các yêu cầu và nộp bài theo hướng dẫn chi tiết bên dưới.

### II. Yêu cầu

Sinh được thực hiện đồ án trên Google Colab với ngôn ngữ lập trình Python3 và nộp notebook sau khi hoàn thành.

#### a) Câu 1 (2.0 điểm): Constraint Satisfaction

**YC1\_1:** Sinh viên cài đặt lớp đối tượng **EightQueenSolver** để giải quyết bài toán 8-Hậu bằng thuật toán **backtracking**.

Mô tả lớp **EightQueenSolver**:

<b>EightQueenSolver</b>
+ EightQueenSolver()
+ solve(): void

trong đó, hàm solve() chạy thuật toán **backtracking** và in kết quả ra màn hình dạng ma trận hai chiều với “Q” là quân hậu và “.” là ô trống.

*Sinh viên tùy ý thêm vào lớp đối tượng các thuộc tính và phương thức hỗ trợ.*

*Sinh viên viết một đoạn chương trình để minh họa hoạt động của lớp đối tượng.*

**YC1\_2:** Sinh viên cài đặt lớp đối tượng **NQueenSolver** để giải quyết bài toán N-Hậu bằng thuật toán **backtracking** trong trường hợp tổng quát với N là kích thước của bàn cờ (N dòng, N cột).

Mô tả lớp **NQueenSolver**:

NQueenSolver
- N: int
+ NQueenSolver(N: int)
+ solve(): void

trong đó, hàm solve() chạy thuật toán **backtracking** và in kết quả ra màn hình dạng ma trận hai chiều với “Q” là quân hậu và “.” là ô trống. Trường hợp không có đáp án, hàm solve() in ra màn hình dòng chữ “UNSOLVABLE”.

*Sinh viên tùy ý thêm vào lớp đối tượng các thuộc tính và phương thức hỗ trợ.*

*Sinh viên viết một đoạn chương trình để minh họa hoạt động của lớp đối tượng.*

### b) Câu 2 (2.0 điểm): Adversarial Search

Sinh viên cài đặt lớp **MinimaxDecision** theo mô tả sau.

MinimaxDecision	Node
- root: Node - terminalStates: dict - successors: dict	- identifier: str - value: int
+ MinimaxDecision() + read(filename: str): void + run(): void + print(): void	+ Node() + Node(identifier: str) + Node(identifier: str, value: int) + __str__(): str

**YC2\_1:** Cài đặt hai lớp đối tượng được mô tả, trong đó

- **terminalStates** là một dictionary,
  - key là một Node identifier, ví dụ “n0”, “n1”.
  - value là một số nguyên, ví dụ 10, 20, 30.
- **successors** là dictionary
  - key là một Node object
  - value là một list chứa các Node object,
  - Ví dụ: n0:[n3, n5, n7] nghĩa là n3, n5, n7 là successors của n0.

- Hàm **read()** nhận vào một filename là đường dẫn tới tệp dữ liệu mô tả cấu trúc cây như sau.
  - Dòng đầu tiên chứa hai số nguyên E và L. E là số lượng nhánh, L là số lượng terminal state/node.
  - E dòng tiếp theo, mỗi dòng chứa hai chuỗi dạng “a b” với ý nghĩa b là successor của a.  
Ví dụ: **n1 n2** thì n2 là successor của n1.
  - L dòng tiếp theo, mỗi dòng chứa một chuỗi và một số nguyên dạng “n v” với ý nghĩa v là giá trị utility của n.  
Ví dụ: **n20 30** thì node n20 có utility là 30.
  - “n00” là node root

**YC2\_2:** Cài đặt hàm **print()** để in ra toàn bộ các node trên cây. Mỗi node cần in ra hai thông tin dạng (identifier, value). Sinh viên cài đặt một hàm **đệ quy** cho thao tác này.

Gợi ý: có thể dựa theo thuật toán **backtracking** để in.

**YC2\_3:** Cài đặt hàm **run()** để tìm và lưu lại value cho tất cả các node trên cây theo thuật toán **Minimax Decision**.

*Sinh viên tùy ý thêm vào lớp đối tượng các thuộc tính và phương thức hỗ trợ.*

*Sinh viên viết một đoạn chương trình để minh họa hoạt động của lớp đối tượng.*

### c) Câu 3 (2.0 điểm): Logical Agents

Sinh viên tìm hiểu và sử dụng thư viện Glucose3 để giải bài tám hậu sử dụng propositional logic. Cài đặt PySAT theo lệnh sau.

```
pip install python-sat==0.1.7.dev12
```

Tài liệu về Glucose3: <https://pysathq.github.io/docs/html/api/solvers.html>

**YC3\_1:** Sinh viên cài đặt lớp đối tượng **EightQueenSolver** để giải quyết bài toán 8-Hậu bằng thư viện **Glucose3** với CNF clauses.

Mô tả lớp **EightQueenSolver**:

<b>EightQueenSolver</b>
+ EightQueenSolver()

+ solve(): void
-----------------

trong đó, hàm solve() chạy thuật toán và in kết quả ra màn hình dạng ma trận hai chiều với “Q” là quân hậu và “.” là ô trống.

*Sinh viên tùy ý thêm vào lớp đối tượng các thuộc tính và phương thức hỗ trợ.*

*Sinh viên viết một đoạn chương trình để minh họa hoạt động của lớp đối tượng.*

**YC3\_2:** Sinh viên cài đặt lớp đối tượng **NQueenSolver** để giải quyết bài toán N-Hậu bằng thư viện **Glucose3** với CNF clauses trong trường hợp tổng quát với N là kích thước của bàn cờ (N dòng, N cột).

Mô tả lớp **NQueenSolver**:

<b>NQueenSolver</b>
- N: int
+ NQueenSolver(N: int)
+ solve(): void

trong đó, hàm solve() chạy thuật toán và in kết quả ra màn hình dạng ma trận hai chiều với “Q” là quân hậu và “.” là ô trống. Trường hợp không có đáp án, hàm solve() in ra màn hình dòng chữ “UNSOLVABLE”.

*Sinh viên tùy ý thêm vào lớp đối tượng các thuộc tính và phương thức hỗ trợ.*

*Sinh viên viết một đoạn chương trình để minh họa hoạt động của lớp đối tượng.*

**Gợi ý:** trường hợp bàn cờ 4 x 4

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

- Gán vào mỗi ô trên bàn cờ một số nguyên dương (chính là một propositional symbol)
- Duyệt qua từng ô, thành lập các sentence mô tả ràng buộc. Ví dụ

ô 1 có hậu khi và chỉ khi 2, 3, 4 không có hậu	$1 \Leftrightarrow \neg 2 \wedge \neg 3 \wedge \neg 4$	(s1)
--	--	------

ô 1 có hậu khi và chỉ khi 6, 11, 16 không có hậu	$1 \Leftrightarrow -6 \wedge -11 \wedge -16$	(s2)
ô 1 có hậu khi và chỉ khi 5, 9, 13 không có hậu	$1 \Leftrightarrow -5 \wedge -9 \wedge -13$	(s3)

- Chuyển tất cả các sentence s1, s2, s3 thành dạng CNF clauses.
- Sau khi xử lý toàn bộ ô, các được danh sách các CNF clauses thì dùng thư viện Glucose3 để giải.

#### d) Câu 4 (2.0 điểm): Machine Learning

Sinh viên nghiên cứu và sử dụng thư viện

- Decision Tree: <https://scikit-learn.org/stable/modules/tree.html>
- Naïve Bayes: [https://scikit-learn.org/stable/modules/naive\\_bayes.html](https://scikit-learn.org/stable/modules/naive_bayes.html)
- k-NN: <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>

để xây dựng ba mô hình phân loại chữ số viết tay trong tập MNIST.

Sinh viên sử dụng đoạn mã sau để download tập training và test.

```
from keras.datasets import mnist
(train_X, train_y), (test_X, test_y) = mnist.load_data()
```

#### YC4\_1: Xây dựng mô hình Decision Tree

- Học dữ liệu trên tập training
- Tính độ chính xác trên tập training và tập test
- Lưu mô hình xuống tập tin
- Load mô hình lên từ tập tin
- Chạy inference (tính prediction) cho ít nhất 05 input samples.
- Trực quan hoá cây với hàm `tree.plot_tree()`.

#### YC4\_2: Xây dựng mô hình Naïve Bayes classifier

- Học dữ liệu trên tập training
- Tính độ chính xác trên tập training và tập test
- Lưu mô hình xuống tập tin
- Load mô hình lên từ tập tin
- Chạy inference (tính prediction) cho ít nhất 05 input samples.

**YC4\_3: Xây dựng mô hình k-NN**

- Học dữ liệu trên tập training
- Tính độ chính xác trên tập training và tập test
- Lưu mô hình xuống tập tin
- Load mô hình lên từ tập tin
- Chạy inference (tính prediction) cho ít nhất 05 input samples.

**YC4\_4:** Sử dụng thư viện matplotlib.pyplot để vẽ biểu đồ cột đôi thể hiện độ chính xác trên tập training và test cho ba mô hình ở trên.

**e) Câu 5 (2.0 điểm): Báo cáo**

- Sinh viên viết báo cáo kết quả đồ và quay video thuyết trình. **KHÔNG CÓ MẪU THUYẾT TRÌNH, NHÓM SINH VIÊN TỰ TỔ CHỨC NỘI DUNG.**
- Các thông tin tối thiểu cần có.
  - Danh sách sinh viên: MSSV, Họ tên, Email, Phân công công việc, Mức độ hoàn thành.
  - Tóm tắt cách xử lý từng yêu cầu, nên diễn đạt bằng mã giả/sơ đồ.
  - HẠN CHẾ TỐI ĐA NHÚNG MÃ NGUỒN THÔ VÀO BÀI THUYẾT TRÌNH.
  - Các nội dung tìm hiểu cần trình bày cô đọng, có ví dụ trực quan.
  - Thuận lợi và khó khăn trong đề tài.
  - Bảng tự đánh giá mức độ hoàn thành các yêu cầu.
  - Tài liệu trích dẫn ghi theo định dạng IEEE.
- Yêu cầu về định dạng: tỷ lệ slide 4x3, hạn chế dùng nền tối/màu sắc vì máy chiếu mờ, đảm bảo khi in bài thuyết trình dạng trắng đen thì các nội dung vẫn rõ ràng.
- Thời lượng tối đa cho phần thuyết trình là **05 phút**.
- *Cách thức quay video thuyết trình được đính kèm ở phụ lục.*

**III. Hướng dẫn nộp bài**

- Tạo thư mục với tên theo cú pháp

<MSSV1>\_<MSSV2>\_<MSSV3>\_<MSSV4>\_<MSSV5>

trong đó gồm:

- **source:** thư mục mã nguồn chứa tệp .ipynb
- **presentation.pdf:** bài thuyết trình.
- **video.txt:** chứa đường dẫn tới video thuyết trình.
- Nén thư mục thành tệp zip và nộp theo deadline.

#### IV. Quy định

- **Nhóm sinh viên nộp trễ hạn bị 0.0 điểm toàn nhóm.**
- **Sai sót mã số sinh viên nào trong tên tệp nộp bài thì sinh viên tương ứng bị 0.0 điểm.**
- **Thiếu sót các tài liệu được yêu cầu trong tệp nộp bài sẽ bị trừ tối thiểu 50% điểm phần thuyết trình.**
- **Mọi hành vi sao chép code trên mạng, chép bài bạn hoặc cho bạn chép bài nếu bị phát hiện đều sẽ bị điểm 0.0.**
- **Nếu bài làm của sinh viên có dấu hiệu sao chép trên mạng hoặc sao chép nhau, sinh viên sẽ được gọi lên phòng vấn code riêng để chứng minh bài làm là của mình.**

-- HẾT --

# HƯỚNG DẪN VIDEO THUYẾT TRÌNH *INSTRUCTIONS*

## ***FOR RECORDING PRESENTATION VIDEO***

### **I. Mục tiêu/Objectives**

- Nhóm sinh viên thực hiện quay video thuyết trình để báo cáo nội dung đề án/đề tài.  
*Student groups record a video to present your project/topic.*
- Hình thức, công cụ, thời lượng được mô tả chi tiết ở mục tiếp theo.  
*Formation, tools, and duration are described in the next section.*

### **II. Yêu cầu/Requirements**

- Công cụ: Zoom/Google Meet  
*Tools: Zoom/Google Meet*
- Thời lượng: tối đa 05 phút  
*Duration: 05 minutes (no longer than 05 minutes)*
- Hình thức:  
*Formation*
  - Nhóm sinh viên tạo một meeting để tham gia thuyết trình.  
*Student groups create a meeting to present your work.*
  - Đặt tên hiển thị theo dạng <MSSV>\_<Họ tên>,  
*Set the display name as <Student ID>\_<Full Name>*  
ví dụ 52200001\_Nguyễn Văn A  
*for example, 52200001\_Nguyen Van A*
  - Tất cả thành viên phải bật camera trong toàn bộ buổi thuyết trình.  
*Every member must turn on your camera during the presentation.*
  - Sinh viên trình chiếu bài thuyết trình nhưng phải đảm bảo hiển thị đầy đủ khuôn mặt của các thành viên còn lại.



*Students show your presentation but ensuring to display all member facial thumbnails.*

- Các sinh viên thay phiên trình bày các nội dung.

*Students share the role of presenting.*

- Bài thuyết trình được quay lại, đảm bảo chất lượng hình ảnh và âm thanh.

*The presentation is recorded completely.*

### III. Hướng dẫn nộp bài/Submission Instructions

- Video tải lên Youtube và đặt ở chế độ “unlisted”, tuyệt đối không để dạng “public”.

*The recording is uploaded to Youtube with the “unlisted” sharing option. Do not share with the “public” option.*

- Đặt tên theo cú pháp

*Rename the recording as below*

<Năm học>-<Học kỳ>-<Môn học>-<Tên nhóm>

<Year>-<Semester>-<Course>-<Group Name>

trong đó gồm:

- <Năm học> theo dạng YYYY, ví dụ 2223, 2324, 2425  
*<Year> in form of YYYY, for example, 2223, 2324, 2425*
- <Học kỳ> là “HK1” hoặc “HK2”  
*<Semester> is “HK1” (Term 1) or “HK2” (Term 2)*
- <Môn học> là “AI” (NM Trí tuệ Nhân tạo) hoặc “MMDS” (Xử lý Dữ liệu lớn)  
*<Course> is “AI” (Introduction to AI) or “MMDS” (Mining Massive Datasets)*
- <Tên nhóm> theo tên đã đăng ký  
*<Group Name> as in registration*

- Một thành viên nhóm đại diện nộp đường dẫn theo deadline được cho.

*Only one representative student submits the video URL by the deadline.*

-- HẾT --

-- END --