# BÁO CÁO CUỐI KỲ

**Môn học: CS2205 - PHƯƠNG PHÁP LUẬN NCKH**
**Lớp:  CS2205.CH183**
**GV: PSG. TS. Lê Đình Duy**

# TỰ ĐỘNG HOÁ TẠO HÀM PHẦN THƯỞNG BẰNG MÔ HÌNH NGÔN NGỮ LỚN TRONG HỌC TĂNG CƯỜNG

Phạm Huỳnh Phúc - 230101015

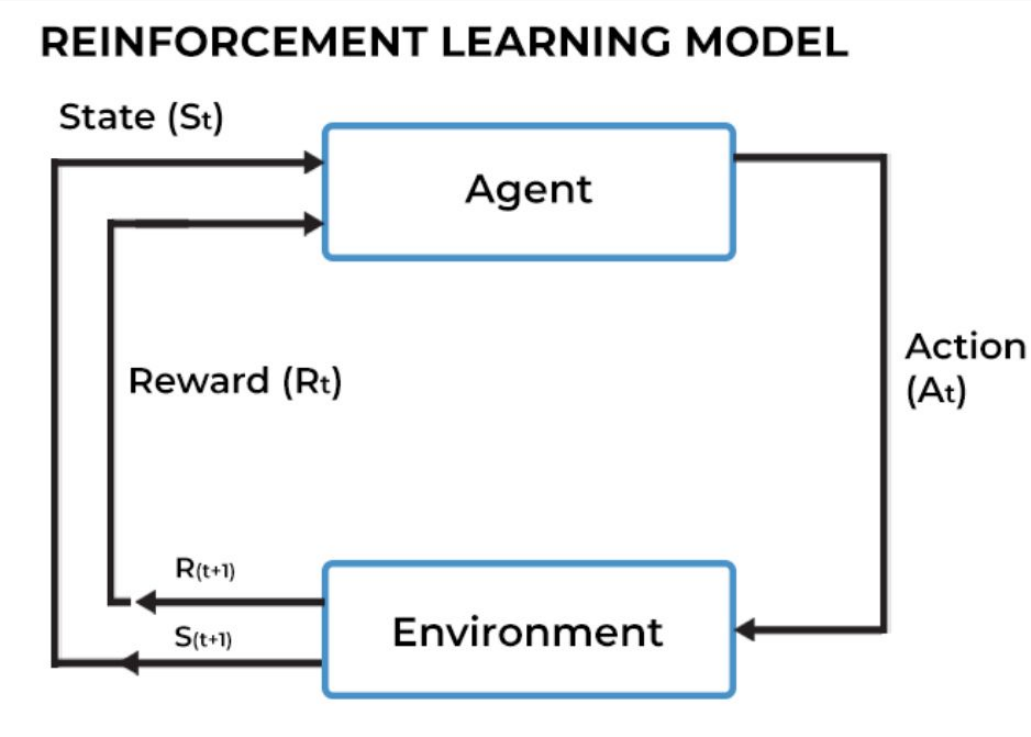
# Tóm tắt

- Lớp: CS2205.CH2023.01
- Link Github của nhóm: https://github.com/phchynhuu/CS2205.CS2205/blob/master/PhamHuynhPhuc-CS2205.NOV2024.Slides.pdf
- Link YouTube video:
  - https://youtu.be/K7y9DXtyBOg
- Ảnh + Họ và Tên của các thành viên
  - Phạm Huỳnh Phúc - 230101015

# Giới thiệu

- Bối cảnh
  - Học tăng cường (RL) đã chứng minh thành công trong nhiều lĩnh vực
  - Thiết kế hàm reward hiệu quả vẫn là thách thức lớn
  - Phương pháp truyền thống đòi hỏi chuyên môn cao và tốn nhiều công sức
- Động lực nghiên cứu
  - LLMs phát triển mạnh mẽ với khả năng lập trình và suy luận vượt trội
  - Công trình EUREKA chứng minh tiềm năng của LLMs trong tạo hàm reward
  - Cần giải pháp tự động hóa thiết kế hàm reward để giảm rào cản kỹ thuật
- Đề xuất
  - Tận dụng khả năng mã hóa của LLMs để tự động thiết kế hàm reward
  - Kết hợp phương pháp few-shot, chain-of-thought và tiến hóa lặp
  - Đánh giá hiệu quả qua so sánh với hàm reward thiết kế thủ công

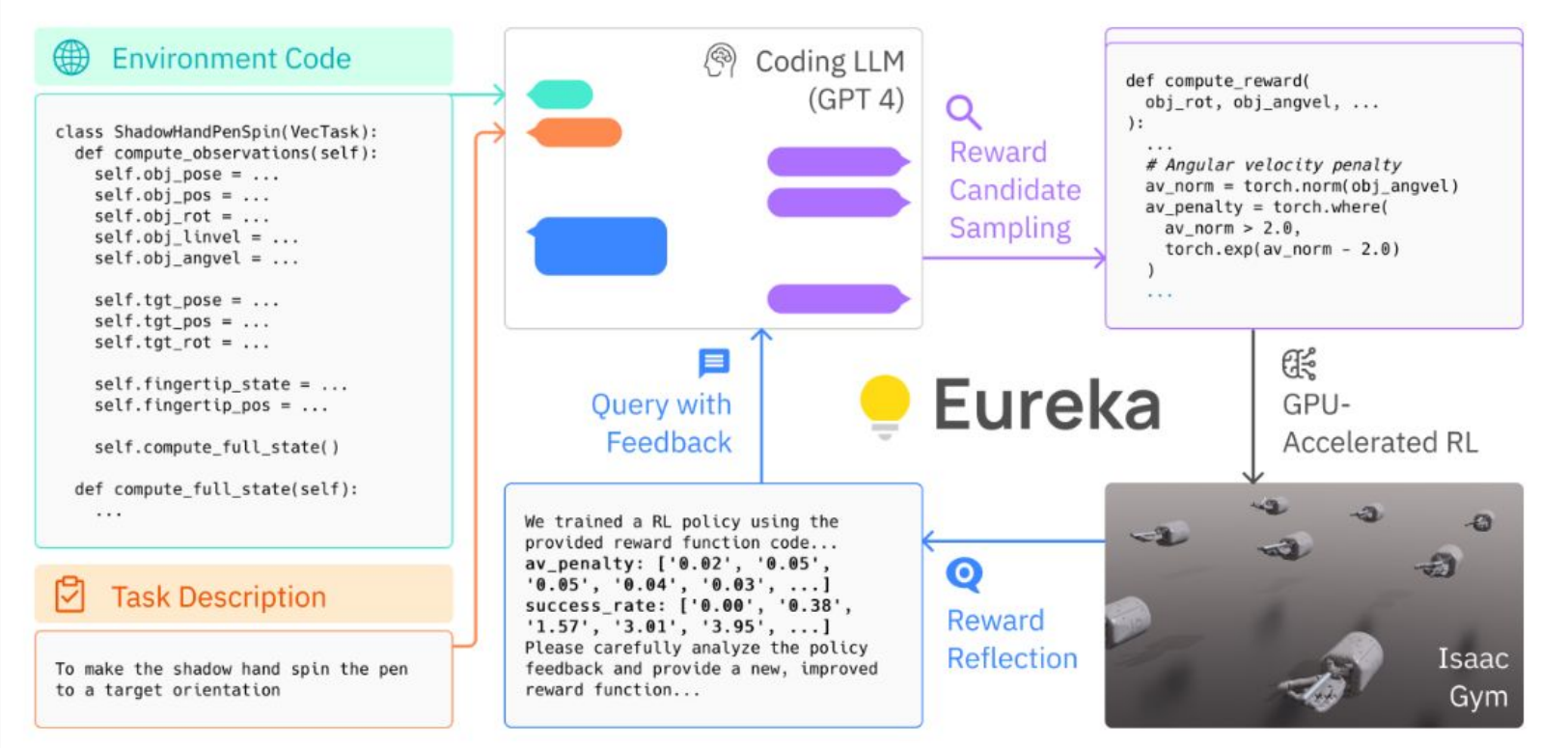
# Giới thiệu


REINFORCEMENT LEARNING MODEL
State (St)
Agent
Reward (Rt)
Action
(At)
R(t+1)
S(t+1)
Environment

# Mục tiêu

- Phát triển và kiểm nghiệm phương pháp tự động tạo hàm reward bằng mô hình ngôn ngữ lớn (LLMs) cho các bài toán học tăng cường.
- So sánh và đánh giá hiệu suất của hàm Reward(Rt) được tạo tự động bằng các huấn luyện agent với các môi trường benchmark cụ thể và đo lường khoảng cách hiệu năng giữa hàm do LLMs và con người thiết kế.
- Phân tích đánh giá các yếu tố ảnh hưởng đến chất lượng của hàm Reward được tạo tự động bao gồm độ phức tạp của môi trường, chất lượng mô tả đầu vào, và khả năng của các LLMs khác nhau từ đó đưa ra nhận xét về tiềm năng cũng như giới hạn của phương pháp này.

# Nội dung và Phương pháp



# Kết quả dự kiến

- Chứng minh được hiệu quả của việc tự động tạo hàm reward bằng các mô hình ngôn ngữ lớn có khả năng suy luận.
- So sánh được hiệu suất của việc thiết kế và tối ưu hoá prompt từ các phương pháp khác nhau.
- Cung cấp tới cộng đồng một bộ mã Python phục vụ cho mục đích nghiên cứu và phát triển đề tài và tài liệu hướng dẫn chi tiết.