

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**



---

**BÀI TẬP MÔN TRÍ TUỆ NHÂN TẠO**

---

**BÀI TẬP 4 – Evaluation functions for Minimax/AlphaBeta/Expectimax**

**GV hướng dẫn:**

TS. Lương Ngọc Hoàng

**Sinh viên thực hiện:**

Trương Phúc Trường – 22521587

TP. HCM, ngày 24 tháng 4 năm 2024

---

## Ý tưởng/chiến thuật thiết kế evaluation function

Dựa vào các thông tin em tìm kiếm được trong file *pacman.py* như sau:

- Chiến thắng trò chơi sẽ được **cộng 500** điểm.
- Thua trò chơi sẽ bị **trừ 500** điểm.
- Ăn con ma sẽ **cộng 200** điểm.
- Ăn mỗi chấm thức ăn sẽ **cộng 10** điểm.

Nên em đã sử dụng các đặc trưng sau để ước lượng giá trị trạng thái:

### *1. Khoảng cách từ Pacman tới chấm thức ăn gần nhất (closest\_food):*

- + Trọng số: **-100**
- + Mục đích: Trọng số âm lớn để Pacman ưu tiên ăn thức ăn gần nhất, tránh tình trạng bị đói thức ăn.

### *2. Khoảng cách từ Pacman tới con ma gần nhất (closest\_ghost):*

- + Trọng số: **-0.2**
- + Mục đích: Trọng số âm nhỏ cho khoảng cách tới ma để không quá thận trọng, cân bằng giữa ăn thức ăn và tránh ma, tránh tình trạng Pacman quá bị động chỉ toàn phòng thủ.

### *3. Khoảng cách từ Pacman tới capsule gần nhất (closest\_capsule):*

- + Trọng số: **-5**
- + Mục đích: Trọng số âm vừa phải cho capsule vì khi ăn capsule sẽ làm các con ma sợ hãi trong thời gian ngắn và có thể tranh thủ thời gian này để ăn con ma.

### *4. Số lượng chấm thức ăn còn lại (remaining\_food):*

- + Trọng số: **-20**
- + Mục đích: Trọng số âm cho số lượng thức ăn còn lại vừa phải để khuyến khích thu gom càng nhiều càng tốt.

**5. Số lượng capsule còn lại (*remaining\_capsules*):**

+ Trọng số: **-30**

+ Mục đích: Tương tự như số lượng thức ăn còn lại.

**6. Thời gian con ma còn sợ hãi (*remaining\_scared\_time*):**

+ Trọng số: **2**

+ Mục đích: Trọng số dương nhỏ cho thời gian ma sợ hãi để khuyến khích Pacman tấn công ma khi chúng sợ.

**7. Khoảng cách từ Pacman tới góc tường gần nhất (*closest\_wall*):**

+ Trọng số: **10**

+ Mục đích: Trọng số dương cho khoảng cách góc tường để đảm bảo an toàn, tránh tình trạng bị dồn vào góc tường sẽ làm thua game.

**8. Số lượng đường đi khả dụng (*available\_paths*):**

+ Trọng số: **5**

+ Mục đích: Trọng số dương cho số đường đi để khuyến khích Pacman di chuyển tự do hơn.

*Công thức lượng giá:*

$$\begin{aligned} \text{evaluation} = & -100 * \text{closest\_food} - 0.2 / \text{closest\_ghost} - 5 / \text{closest\_capsule} \\ & - 20 * \text{remaining\_food} - 30 * \text{remaining\_capsules} \\ & + 2 * \text{remaining\_scared\_time} + 10 / \text{closest\_wall} + 5 * \text{available\_paths} \end{aligned}$$

## **Thực nghiệm**

– Do kết quả của MinimaxAgent và AlphaBetaAgent cho ra là giống nhau nên em đã gom lại thành 1 và số lần từ 1 đến 5 ứng với các random seed từ 22521587+0 → 22521587+4.

– Các thuật toán được chạy với số bước nhìn trước là 3.

*Kết quả chạy thực nghiệm các thuật toán đã cài đặt với **scoreEvaluationFunction***

		<b>MinimaxAgent, AlphaBetaAgent</b>		<b>ExpectimaxAgent</b>	
Layouts	Số lần	Scores	Win	Scores	Win
<b>contestClassic</b>	1	1000	0	-55	0
	2	23	0	-56	0
	3	-361	0	71	0
	4	-248	0	229	0
	5	40	0	2000	1
<b>minimaxClassic</b>	1	-492	0	504	1
	2	511	1	511	1
	3	-496	0	-495	0
	4	511	1	511	1
	5	-492	0	507	1
<b>testClassic</b>	1	526	1	526	1
	2	546	1	546	1
	3	552	1	554	1
	4	488	1	490	1
	5	512	1	512	1
<b>trappedClassic</b>	1	-501	0	532	1
	2	-501	0	532	1
	3	-501	0	-502	0
	4	-501	0	532	1
	5	-501	0	532	1
<b>capsuleClassic</b>	1	-448	0	-450	0
	2	-473	0	-278	0
	3	-515	0	-288	0
	4	-421	0	-272	0
	5	-447	0	-103	0

*Kết quả chạy thực nghiệm các thuật toán đã cài đặt với **newbetterEvaluationFunction***

		<b>MinimaxAgent, AlphaBetaAgent</b>		<b>ExpectimaxAgent</b>	
Layouts	Số lần	Scores	Win	Scores	Win
<b>contestClassic</b>	1	256	0	71	0
	2	-44	0	-46	0
	3	284	0	284	0
	4	-145	0	-117	0

	5	-427	0	-337	0
<b>minimaxClassic</b>	1	-503	0	501	1
	2	512	1	512	1
	3	-496	0	-498	0
	4	512	1	512	1
	5	-503	0	507	1
<b>testClassic</b>	1	526	1	535	1
	2	513	1	548	1
	3	486	1	552	1
	4	497	1	491	1
	5	508	1	511	1
<b>trappedClassic</b>	1	532	1	532	1
	2	532	1	532	1
	3	-502	0	-502	0
	4	532	1	532	1
	5	532	1	532	1
<b>capsuleClassic</b>	1	-459	0	-470	0
	2	-143	0	-143	0
	3	-321	0	-353	0
	4	-450	0	-461	0
	5	65	0	65	0

### *Nhận xét:*

- Ta có thể thấy hầu như ở các trường hợp, thuật toán MinimaxAgent và AlphaBetaAgent có phần tốt hơn.
- Trong các thuật toán, ta có thuật toán Minimax và AlphaBeta tuy là luôn có cùng kết quả thực nghiệm, nhưng ở AlphaBeta sẽ cho thời gian xử lý thấp hơn so với Minimax nếu các nhánh có thứ tự sắp xếp tốt hơn và độ sâu lớn hơn nhờ vào bước cải tiến Pruning so với minimax. Còn ở thuật toán Expectimax, việc thiết kế ước lượng theo giá trị kì vọng, có thể chưa phù hợp trong các layout hiện tại của Pacman, dẫn đến kết quả chưa thực sự tốt so với Minimax và AlphaBeta.