Trần Hồ Phương Nguyên – 23110271

Nguyễn Thị Hoàng Kim – 23110248

Đoàn Quang Khôi – 23110244

Đề tài: TRUY TÌM KHO BÁU

1. Xác Định Trạng Thái

1.1 Trạng thái Đầu (Initial State)

* Vị trí người chơi: (x\_start, y\_start) (ngẫu nhiên trên bản đồ).
* Tài nguyên ban đầu:
  + Nước: 50%
  + Thức ăn: 50%
  + Sức khỏe: 100%
* Bản đồ khảo sát: Chỉ hiển thị khu vực gần người chơi.
* Thời gian trong ngày: Có thể là sáng hoặc tối tùy thiết kế game.

1.2 Trạng thái Đích (Goal State)

* Người chơi tìm thấy kho báu tại vị trí (x\_goal, y\_goal).
* Người chơi phải sống sót (không chết do thiếu nước, thức ăn, hoặc bị thú dữ tấn công).
* Game có thể có nhều kết thúc, bao gồm tìm kho báu và thoát khỏi đảo.
* Thắng game: Tìm được kho báu hoặc trốn thoát thành công.
* Thua game: Chết vì đói, khát, thú dữ hoặc rơi vào bẫy.

2. Cấu trúc lưu trữ trạng thái

Mỗi trạng thái trong game sẽ được biểu diễn dưới dạng class Python:

class State:

def \_\_init\_\_(self, x, y, water, food, health, time, terrain, tools):

self.position = (x, y)

self.resources = {"water": water, "food": food, "health": health}

self.environment = {"time": time, "terrain": terrain}

self.inventory = {"tools": tools}

def \_\_eq\_\_(self, other):

return self.position == other.position

def \_\_hash\_\_(self):

return hash(self.position)

Một trạng thái có thể được biểu diễn dưới dạng tuple hoặc class, gồm:

* Vị trí trên bản đồ: (x, y)
* Lượng tài nguyên: { "nước": %, "thức ăn": %, "sức khỏe": % }
* Trạng thái môi trường: { "thời gian": ngày/đêm, "địa hình": rừng/sông/núi }
* Hành trang: { "công cụ": True/False, "bản đồ": True/False }

A screenshot of a computer

AI-generated content may be incorrect.

3. Ba Thuật Toán Tính Chi Phí

3.1 Uniform Cost Search (UCS) - Chi phí thực tế thấp nhất

* Chi phí = tổng quãng đường + độ khó địa hình.
* Các hình thức địa hình:
  + Cát = 1, Rừng = 2, Núi = 4, Sông = 5.

Công thức:

g(n) = g(n – 1) + cost(địa hình)

3.2 Greedy Best-First Search - Chi phí theo ước lượng

* Chi phí = khoảng cách trực tiếp đến kho báu.
* Dùng khi muốn di chuyển nhanh nhất.

Công thức:

A black and white math equation

AI-generated content may be incorrect.

*3.3 A Search - Kết hợp UCS và Greedy*\*

* Chi phí = tổng quãng đường + khoảng cách ước lượng.
* Tìm đường nhanh và hiệu quả nhất.

Công thức:

f(n)=g(n)+h(n)f(n) = g(n) + h(n)