



تمرین اول

نام درس: سیستم های چندعاملی

استاد درس: دکتر ناصر مزینی

نام: محمد حقیقت

شماره دانشجویی: 403722042

گرایش: هوش مصنوعی

دانشکده: مهندسی کامپیوتر

نیم سال دوم 1403-1404

در تمرین دوم کد های مربوط به محیط و عامل مبتنی بر LLM رو توضیح دادیم و در گزارش این تمرین فقط بخش هایی که تغییر کردند را گزارش می کنیم.

```
# Constants
GRID_SIZE = 15
NUM_BUILDINGS = 7
NUM_CIVILIANS = 8
NUM_TYPE1_FIRES = 5
NUM_TYPE2_FIRES = 3
MAX_STEPS = 50
SENSOR_RANGE = 2
```

در تمرین قبل فقط یک نوع آتش با ثابت  $\text{NUM\_FIRES} = 5$  داشتیم ولی در این بخش دو نوع آتش معرفی شده است:  $\text{NUM\_TYPE1\_FIRES} = 5$  و  $\text{NUM\_TYPE2\_FIRES} = 3$

این تغییر برای افزایش پیچیدگی محیط اضافه شده است. آتش نوع دوم (Type 2 Fire) نیاز به همکاری دو عامل برای خاموش کردن دارد.

```
# Cell types
EMPTY = 0
BUILDING = 1
CIVILIAN = 2
FIRE = 3
AGENT1 = 4
AGENT2 = 5
STATION = 6
LARGE_FIRE = 7
```

تغییر در تعریف انواع سلول ها:

اضافه شدن  $\text{AGENT1}$ ,  $\text{AGENT2}$  و  $\text{LARGE\_FIRE}$  و تغییر شماره گذاری (مثلا  $\text{STATION}$  از ۵ به ۶ و اضافه شدن  $\text{LARGE\_FIRE}$  به عنوان ۷)

این تغییر برای پشتیبانی از دو عامل و نوع جدید آتش (Type 2 Fire) انجام شده است.  $\text{AGENT1}$  و  $\text{AGENT2}$  نشان دهنده دو عامل مجزا هستند و  $\text{LARGE\_FIRE}$  آتش نوع دوم را نشان می دهد.

```
# Custom color mapping
COLORS = {
    EMPTY: "lightgray",
    BUILDING: "yellow",
    CIVILIAN: "blue",
    FIRE: "red",
    AGENT1: "green",
    AGENT2: "cyan",
    STATION: "purple",
    LARGE_FIRE: "orange"
}
```

به‌روزرسانی رنگ‌ها و نقشه‌های رنگی:

کد جدید رنگ‌های جدید cyan برای AGENT2 و orange برای LARGE\_FIRE و برچسب‌های جدید در COLOR\_LEGEND اضافه کرده است.

این تغییرات برای نمایش بصری دو عامل و آتش نوع دوم در نقشه‌ها انجام شده است. همچنین، تعداد رنگ‌ها در FULL\_CMAP از ۶ به ۸ افزایش یافته است.

```
def setup_environment(self):
    self.grid[0, 0] = STATION
    self.grid[14, 14] = STATION
    for _ in range(NUM_BUILDINGS):
        self.place_random_entity(BUILDING)
    for _ in range(NUM_CIVILIANS):
        self.place_random_entity(CIVILIAN)
    for _ in range(NUM_TYPE1_FIRES):
        self.place_random_entity(FIRE)
    for _ in range(NUM_TYPE2_FIRES):
        self.place_random_entity(LARGE_FIRE)
```

تغییرات در تنظیمات اولیه محیط:

اضافه شدن LARGE\_FIRE در کنار FIRE در تابع setup\_environment.

این تغییر برای پشتیبانی از آتش نوع دوم است که نیاز به همکاری دو عامل برای خاموش کردن دارد.

```
def spread_fire(self):
    fire_positions = list(zip(*np.where((self.grid == FIRE) | (self.grid == LARGE_FIRE))))
    if not fire_positions:
        return
    for x, y in fire_positions[:]:
        directions = [(0, 1), (0, -1), (1, 0), (-1, 0)]
        random.shuffle(directions)
        for dx, dy in directions:
            new_x, new_y = x + dx, y + dy
            if (0 <= new_x < GRID_SIZE and 0 <= new_y < GRID_SIZE and
                self.grid[new_x, new_y] not in [FIRE, LARGE_FIRE, BUILDING, STATION, AGENT1, AGENT2]):
                if self.grid[new_x, new_y] == CIVILIAN:
                    self.score -= 100
                    print(f"Civilian died in fire! Score: {self.score}")
                self.grid[new_x, new_y] = FIRE # All fires spread as Type 1
                break
```

گسترش آتش:

تابع `spread_fire` حالا هر دو نوع آتش (`FIRE` و `LARGE_FIRE`) را در نظر می‌گیرد، اما هنگام گسترش، فقط آتش نوع اول (`FIRE`) ایجاد می‌شود.

این محدودیت برای ساده‌سازی گسترش آتش اعمال شده است، اما ممکن است باعث شود آتش نوع دوم به‌صورت غیرمستقیم از طریق گسترش آتش نوع اول افزایش یابد.

```
def is_game_over(self):
    return (self.steps >= MAX_STEPS or
            (self.type1_fires_extinguished == NUM_TYPE1_FIRES and
             self.type2_fires_extinguished == NUM_TYPE2_FIRES and
             self.civilians_rescued == NUM_CIVILIANS))
```

مدیریت امتیازات و پایان بازی:

متغیرهای `type1_fires_extinguished` و `type2_fires_extinguished` اضافه شده‌اند، و شرط پایان بازی شامل خاموش شدن هر دو نوع آتش است.

این تغییر برای تطبیق با وجود دو نوع آتش و نیاز به مدیریت جداگانه آن‌ها انجام شده است.

```
def apply_action(self, x, y, action, last_cell_content, agent_id, other_agent_pos):
    """
    Apply the specified action at position (x, y) for the given agent.
    Returns True if the action was successful, False otherwise.
    """
    if action == "extinguish fire" and last_cell_content == FIRE:
        self.update_cell(x, y, EMPTY)
        self.type1_fires_extinguished += 1
        self.score += 10
        print(f"Agent {agent_id} extinguished Type 1 fire! Score: {self.score}")
        return True
    elif action == "extinguish fire" and last_cell_content == LARGE_FIRE:
        # Check if both agents are on the same cell and both attempt to extinguish
        if other_agent_pos == (x, y):
            self.update_cell(x, y, EMPTY)
            self.type2_fires_extinguished += 1
            self.score += 25
            print(f"Agents cooperatively extinguished Type 2 fire! Score: {self.score}")
            return True
        return False
```

تابع `apply_action`:

تابع `apply_action` حالا `agent_id` و `other_agent_pos` را به عنوان ورودی می‌گیرد و برای خاموش کردن `LARGE_FIRE` نیاز به حضور هر دو عامل در یک سلول دارد. همچنین، امتیاز خاموش کردن `LARGE_FIRE` برابر با ۲۵ است (در مقابل ۱۰ برای `FIRE`).

این تغییر برای پشتیبانی از همکاری بین دو عامل در خاموش کردن آتش نوع دوم طراحی شده است.

```
class LLMAgent:
    def __init__(self, environment, email, password, agent_id, shared_map):
        self.env = environment
        self.agent_id = agent_id
        self.agent_type = AGENT1 if agent_id == 1 else AGENT2
        self.pos = None
        self.shared_map = shared_map
        self.discovered_map = np.full((GRID_SIZE, GRID_SIZE), -1)
        self.carrying_civilian = False
        self.known_fires = [] # Type 1 fires
        self.known_large_fires = [] # Type 2 fires
        self.known_civilians = []
        self.known_buildings = []
        self.stations = [(0, 0), (14, 14)]
        self.llm_responses = []
        self.chatbot = self.initialize_chatbot(email, password)
        self.last_cell_content = EMPTY
        self.place_agent()
        self.update_discovered_map()
```

تغییرات در کلاس `LLMAgent`

پشتیبانی از دو عامل:

دو عامل (AGENT1 و AGENT2) با agent\_id و agent\_type تعریف شده‌اند. همچنین، یک shared\_map برای اشتراک اطلاعات بین دو عامل اضافه شده است.

این تغییر برای این است که در آن دو عامل می‌توانند اطلاعات نقشه را به اشتراک بگذارند و با هم همکاری کنند.

```
def update_discovered_map(self):
    x, y = self.pos
    for i in range(max(0, x - SENSOR_RANGE), min(GRID_SIZE, x + SENSOR_RANGE + 1)):
        for j in range(max(0, y - SENSOR_RANGE), min(GRID_SIZE, y + SENSOR_RANGE + 1)):
            cell = self.env.get_grid()[i, j]
            self.discovered_map[i, j] = cell
            self.shared_map[i, j] = cell
            pos = (i, j)
            if cell == FIRE and pos not in self.known_fires:
                self.known_fires.append(pos)
            elif cell == LARGE_FIRE and pos not in self.known_large_fires:
                self.known_large_fires.append(pos)
            elif cell == CIVILIAN and pos not in self.known_civilians:
                self.known_civilians.append(pos)
            elif cell == BUILDING and pos not in self.known_buildings:
                self.known_buildings.append(pos)
            if cell != FIRE and pos in self.known_fires:
                self.known_fires.remove(pos)
            if cell != LARGE_FIRE and pos in self.known_large_fires:
                self.known_large_fires.remove(pos)
            if cell != CIVILIAN and pos in self.known_civilians:
                self.known_civilians.remove(pos)
```

مدیریت نقشه مشترک (shared\_map):

نقشه مشترک (shared\_map) به‌عنوان ورودی به سازنده کلاس LLMAgent اضافه شده و در تابع update\_discovered\_map به‌روزرسانی می‌شود.

این ویژگی امکان هماهنگی بهتر بین دو عامل را فراهم می‌کند، زیرا هر دو به اطلاعات یکسانی از محیط دسترسی دارند.

متغیر known\_large\_fires برای ردیابی موقعیت‌های آتش نوع دوم اضافه شده است.

این تغییر برای تمایز بین آتش نوع اول و دوم و برنامه‌ریزی اقدامات مربوط به آن‌ها لازم است.

```
[AGENT STATUS]
Agent ID: {self.agent_id}
Grid size: ({GRID_SIZE}, {GRID_SIZE})
Agent Position: ({self.pos[0]}, {self.pos[1]})
Other Agent Position: {other_agent_pos}
Shared Map ({GRID_SIZE}x{GRID_SIZE}, -1=unknown, 0=empty, 1=building, 2=civilian, 3=Type1 fire, 4=agent1, 5=agent2, 6=station, 7=type2 fire):
{shared_map_str}
Carrying Civilian: {'Yes' if self.carrying_civilian else 'No'}
Other Agent Carrying Civilian: {'Yes' if other_agent_carrying else 'No'}
Last Cell Content: {self.last_cell_content} (0=Empty, 1=Building, 2=Civilian, 3=Type1 Fire, 6=Station, 7=type2 fire)
Nearest Station: {nearest_station} (Distance: {distance})
Score: {self.env.score}
Steps Remaining: {MAX_STEPS - self.env.steps}
```

تغییر در تابع generate\_prompt:

اطلاعات مربوط به موقعیت و وضعیت عامل دیگر (other\_agent\_pos و other\_agent\_carrying) به پرامپت اضافه شده است. همچنین، اولویت‌ها به‌روزرسانی شده‌اند تا عامل‌ها از موقعیت یکدیگر با خبر باشند تا بتوانند همکاری لازم را برای انجام اعمال داشته باشند.

این تغییرات برای هماهنگی بهتر بین دو عامل و تصمیم‌گیری مبتنی بر وضعیت یکدیگر انجام شده است.

```
def validate_action(self, action, other_agent_pos):
    x, y = self.pos
    if action == "extinguish fire":
        return (self.last_cell_content == FIRE or
                (self.last_cell_content == LARGE_FIRE and other_agent_pos == (x, y)))
    elif action == "rescue civilian":
        return self.last_cell_content == CIVILIAN and not self.carrying_civilian
    elif action == "deliver civilian":
        return self.last_cell_content == STATION and self.carrying_civilian
    elif action in ["move up", "move down", "move left", "move right"]:
        new_x, new_y = x, y
        if action == "move up":
            new_x -= 1
        elif action == "move down":
            new_x += 1
        elif action == "move left":
            new_y -= 1
        elif action == "move right":
            new_y += 1
        return (0 <= new_x < GRID_SIZE and 0 <= new_y < GRID_SIZE and
                (self.shared_map[new_x, new_y] == -1 or
                 self.shared_map[new_x, new_y] not in [BUILDING]))
    return False
```

اعتبارسنجی و اجرای اقدامات:

تابع validate\_action حالا other\_agent\_pos را بررسی می‌کند تا مطمئن شود آتش نوع دوم فقط در صورت حضور هر دو عامل خاموش می‌شود. تابع execute\_action نیز به‌روزرسانی شده تا shared\_map را تغییر دهد.

این تغییرات برای پشتیبانی از همکاری بین دو عامل و به‌روزرسانی همزمان نقشه مشترک ضروری هستند.

```
def display_maps(env, agent1, agent2, step, performance_metrics):
    fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(14, 5))

    ax1.imshow(env.get_grid(), cmap=FULL_CMAP, interpolation='nearest', vmin=0, vmax=7)
    ax1.set_title(f"Full City Map (Step {step})")
    ax1.axis('off')

    ax2.imshow(agent1.shared_map, cmap=DISCOVERED_CMAP, interpolation='nearest', vmin=-1, vmax=7)
    ax2.set_title("Agents' Shared Discovered Map")
    ax2.axis('off')

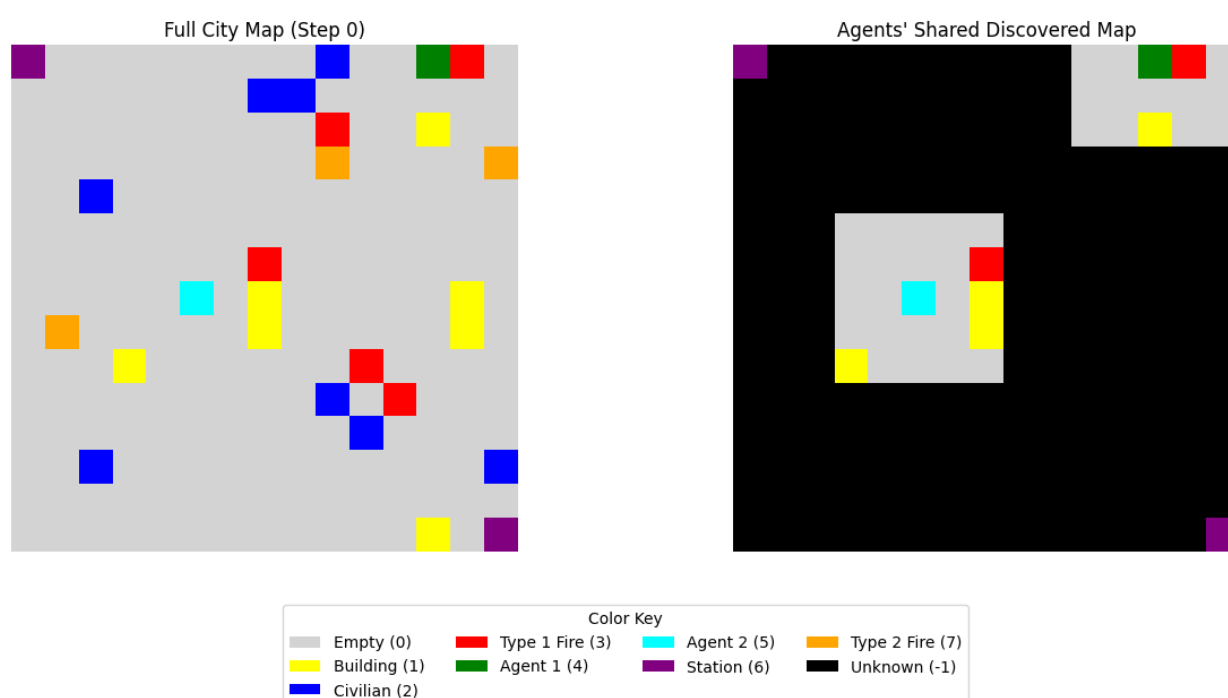
    from matplotlib.patches import Patch
    legend_elements = [Patch(facecolor=color, label=label) for label, color in COLOR_LEGEN.items()]
    fig.legend(handles=legend_elements, loc='upper center', ncol=4, bbox_to_anchor=(0.5, -0.05), title="Color Key")

    plt.tight_layout()
```

تابع `display_maps` حالا دو عامل را می‌پذیرد و نقشه مشترک (`shared_map`) را نمایش می‌دهد. همچنین، معیارهای عملکرد (`performance_metrics`) برای ثبت پوشش اکتشاف و سایر آمارها اضافه شده است.

این تغییرات برای تحلیل بهتر عملکرد سیستم چندعاملی و مقایسه آن با سیستم تک عاملی انجام شده است.

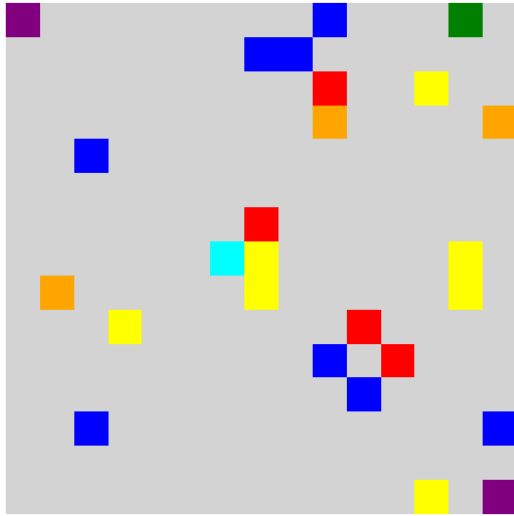
نتایج:



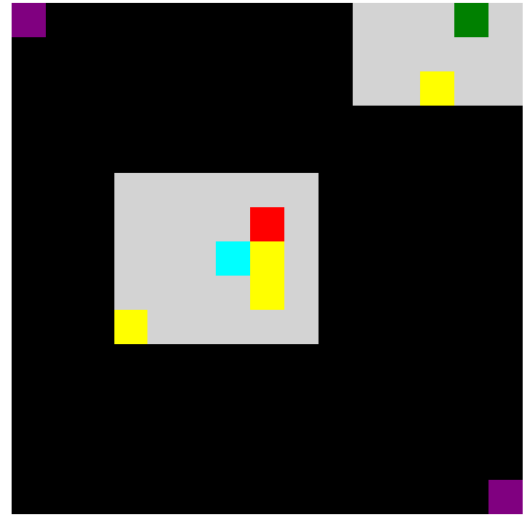
```
Step 0 - Agent 1
LLM Action: move right
Reasoning: The agent should move right because there is a known Type 1 fire at position (0, 13), which is directly to the right of the agent's current position (0, 12). This action fo
Step 0 - Agent 2
LLM Action: move right
Reasoning: The agent is moving towards the nearest known Type 1 fire at position (7, 7) to potentially extinguish it, following priority 5. This action is chosen because it directly ad
```



Full City Map (Step 1)



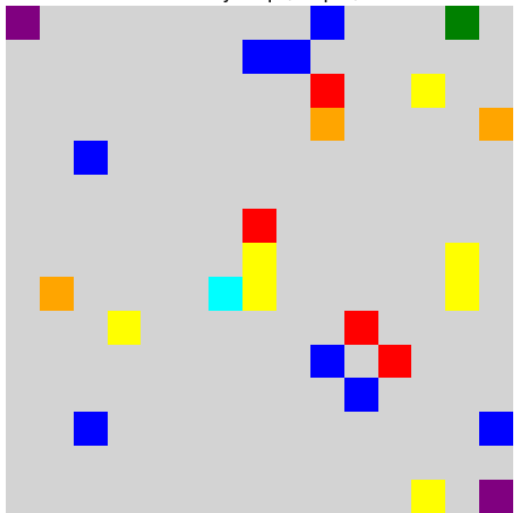
Agents' Shared Discovered Map



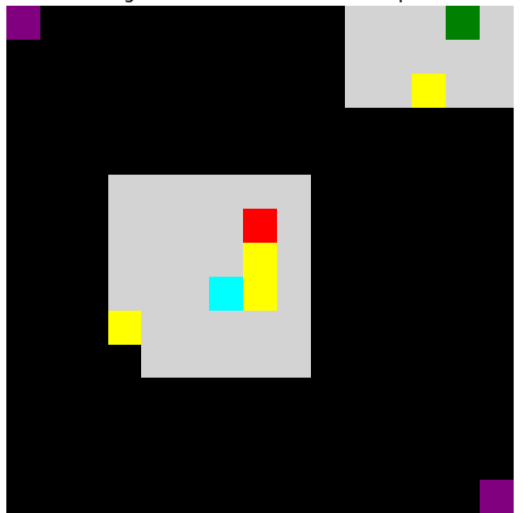
Color Key			
Empty (0)	Type 1 Fire (3)	Agent 2 (5)	Type 2 Fire (7)
Building (1)	Agent 1 (4)	Station (6)	Unknown (-1)
Civilian (2)			

Step 1 - Agent 1  
 LLM Action: extinguish fire  
 Reasoning: The agent is currently on a Type 1 fire, which aligns with priority 3, extinguish a Type 1 fire if on its position. This is the most direct and i  
 Agent 1 extinguished Type 1 fire! Score: 10  
 Step 1 - Agent 2  
 LLM Action: move right  
 Reasoning: The agent is moving towards the nearest known Type 1 fire at position (7, 7) to potentially extinguish it, following priority 5. This action is c  
 Invalid action: move right. Choosing fallback action.  
 Fallback Action: move down

Full City Map (Step 2)



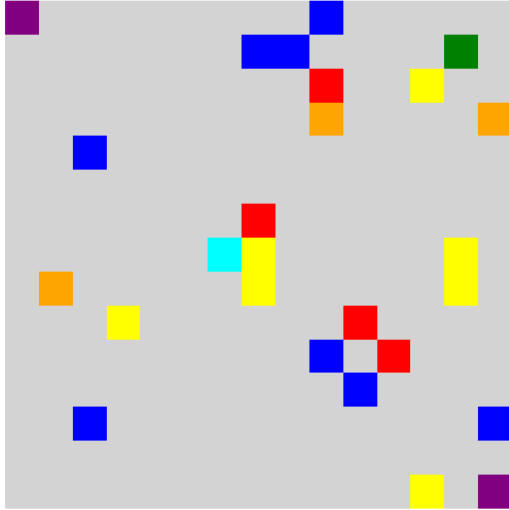
Agents' Shared Discovered Map



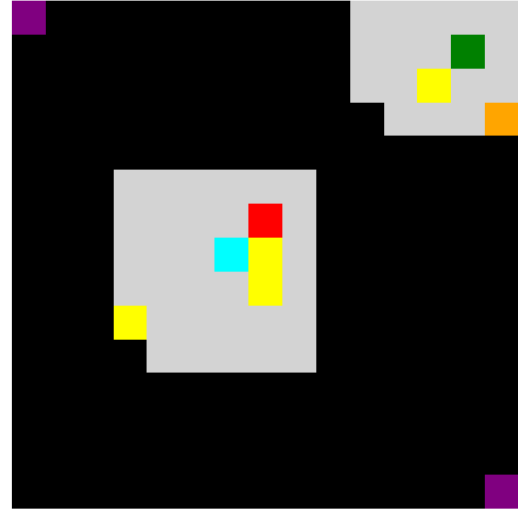
Color Key			
Empty (0)	Type 1 Fire (3)	Agent 2 (5)	Type 2 Fire (7)
Building (1)	Agent 1 (4)	Station (6)	Unknown (-1)
Civilian (2)			

Step 2 - Agent 1  
 LLM Action: move down  
 Reasoning: The agent has just extinguished a Type 1 fire and is now in a position to move towards other potential targets. Moving down to (1, 13) is a safe move that avoids build  
 Step 2 - Agent 2  
 LLM Action: move up  
 Reasoning: The agent is moving up to position (7, 6) to potentially move closer to the Type 1 fire at (7, 7) and to explore the area for civilians, following priorities 5 and 7.

Full City Map (Step 3)



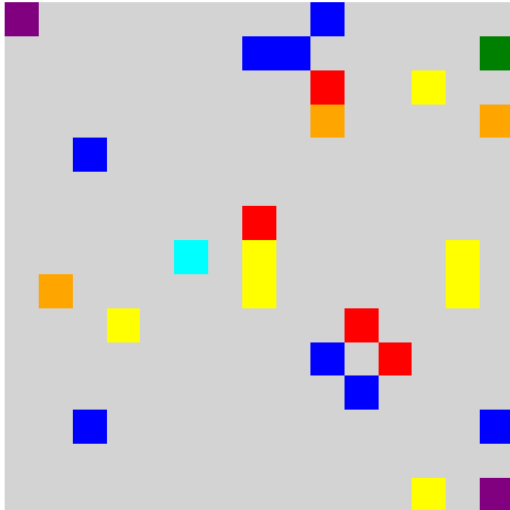
Agents' Shared Discovered Map



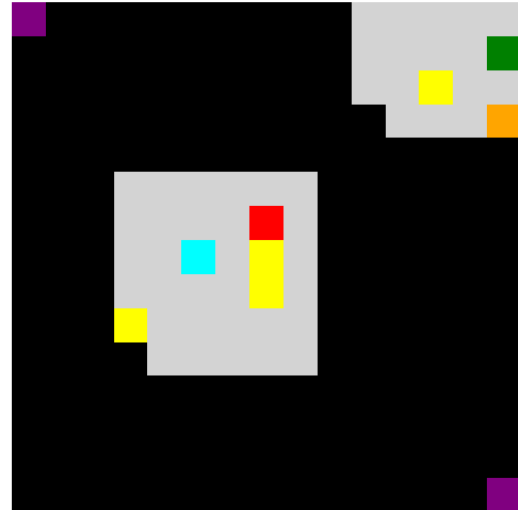
Color Key					
Empty (0)	Type 1 Fire (3)	Agent 2 (5)	Type 2 Fire (7)	Station (6)	Unknown (-1)
Building (1)	Agent 1 (4)				
Civilian (2)					

Step 3 - Agent 1  
 LLM Action: move right  
 Reasoning: The agent should move right because there is a known Type 2 fire at a nearby position, and moving right brings the agent closer to it. This action follows priority 6,  
 Step 3 - Agent 2  
 LLM Action: move right  
 Reasoning: The agent is moving towards the nearest known Type 1 fire at position (7, 7) to potentially extinguish it, following priority 5. This action is chosen because it direc  
 Invalid action: move right. Choosing fallback action.  
 Fallback Action: move left

Full City Map (Step 4)



Agents' Shared Discovered Map



Color Key					
Empty (0)	Type 1 Fire (3)	Agent 2 (5)	Type 2 Fire (7)	Station (6)	Unknown (-1)
Building (1)	Agent 1 (4)				
Civilian (2)					

Step 4 - Agent 1  
 LLM Action: move down  
 Reasoning: The agent should move down because there is a known Type 2 fire nearby, and moving down brings the agent closer to it. This action follows priority 6, moving to  
 Step 4 - Agent 2  
 LLM Action: move right  
 Reasoning: The agent is moving towards the area where the Type 1 fire is located to potentially extinguish it or find other targets, following priority 5. This action is

بقیه نتایج در کد آورده شده است.

برای رفع برخی ایرادات و ابهامات از AI استفاده شده است.