Riphah International University

Artificial Intelligence (AI)

Lab 3

Submitted by: Muhammad Qasim

Sap ID: 37137

Section: BSCS-6A

Submitted To: Mahjabeen

Riphah School of Computing & Innovation Faculty of Computing Riphah International University, Lahore 2023

Lab

```
def print_state(state):
    print("Left:", state["left"])
    print("Right:", state["right"])
    print("Boat:", state["boat"])
    print()

def move(state):
    print_state(state)
    print("Available actions:")
    for i, action in enumerate(actions, 1):
        print(f"{i}. {action}")

    while True:  # Keep asking until a valid input is provided
        choice = input("Choose an action (1-4): ")
        if choice.isdigit() and 1 <= int(choice) <= 4:
            break
        else:
            print("Invalid input. Please enter a number between 1 and 4.")

    action = actions[int(choice) - 1]

    if state["boat"] == "left":
        for item in action.split(", "):
            if item in state["left"]:
                state["left"].remove(item)
                state["right"].add(item)
    else:
        for item in action.split(", "):
            if item in state["right"]:
                state["right"].remove(item)
                state["left"].add(item)

    state["boat"] = "right" if state["boat"] == "left" else "left"

def is_goal_state(state):
    return state == goal_state

def is_valid_move(state, action):
    new_state = state.copy()
    if new_state["boat"] == "left":
        for item in action.split(", "):
            if item not in new_state["left"]:
```

```
                return False
            new_state["left"].remove(item)
            new_state["right"].add(item)
    else:
        for item in action.split(", "):
            if item not in new_state["right"]:
                return False
            new_state["right"].remove(item)
            new_state["left"].add(item)
    return is_valid(new_state) and not goat_eats_corn(new_state) and not
goat_eats_wolf(new_state)

def goat_eats_corn(state):
    return "goat" in state["left"] and "corn" in state["left"] and "farmer" not
in state["left"]

def goat_eats_wolf(state):
    return "goat" in state["left"] and "wolf" in state["left"] and "farmer" not
in state["left"]

initial_state = {"left": {"farmer", "goat", "corn", "wolf"}, "right": set(),
"boat": "left"}

goal_state = {"left": set(), "right": {"farmer", "goat", "corn", "wolf"}, "boat":
"right"}

actions = [
    "farmer",
    "farmer, goat",
    "farmer, corn",
    "farmer, wolf"
]

current_state = initial_state.copy()

while not is_goal_state(current_state):
    move(current_state)
    if goat_eats_corn(current_state):
        print("Oops! The goat ate the corn. You lose!")
        break
    elif goat_eats_wolf(current_state):
        print("Oops! The goat ate the wolf. You lose!")
        break

if is_goal_state(current_state):
```

```
    print("Congratulations! You have successfully moved all characters to the
right side.")
```

OutPut: