

1. Description of the Project (5%)

In this project, we developed a wildfire emergency management system using Python. The project's main objective was to simulate real-world wildfire situations, manage fire alerts, dispatch emergency vehicles, and coordinate evacuation using Queues and Deques. It applies Object-Oriented Programming (OOP) concepts like encapsulation, classes, and methods.

2. Significance of the Project (5%)

Wildfires are a serious natural disaster, and fast response saves lives and property. This project is useful as it performs key actions like receiving alerts, dispatching resources, and evacuating people which are crucial during wildfires. It's novel because these steps use simple data structures learned in the course (Queue and Deque) and use OOP to organize the code.

3. Installation and Usage Instructions (5%)

- **Installation:**
Just copy the Python code and paste it into any ide that supports python (Ex collab)
Make sure Python 3 is installed.
- **Usage:**
Run the file to simulate receiving fire alerts, dispatching emergency vehicles, and managing evacuation queues.

4. Code Structure (5%)

Flow:

FireZone --> FireAlertSystem --> EmergencyResponse --> EvacuationQueue

Main classes:

- Queue and Deque: Handle the order of alerts, vehicles, evacuees.
- FireZone: Represents each fire alert zone.
- FireAlertSystem: Receives and processes fire alerts.
- EmergencyVehicle and EmergencyResponse: Manage vehicles.

- EvacuationQueue: Manages evacuees.

5. Functionalities and Test Results (15%)

Functionalities:

- Add fire alerts to the alert system using Queue.
- Process fire alerts one by one (FIFO order).
- Add emergency vehicles to a Deque.
- Dispatch vehicles (remove from front).
- Add evacuees to evacuation queue.
- Evacuate evacuees in order.

```
# --- Main Code Example ---
# Fire zones representing different fire zones with varying intensities and locations
zone1 = FireZone(101, 8, "Forest area A")
zone2 = FireZone(102, 5, "Forest area B")
zone3 = FireZone(103, 9, "Mountain area C")

# Fire alert system to receive and process fire alerts
alert_system = FireAlertSystem()
alert_system.receiveAlert(zone1)
alert_system.receiveAlert(zone2)
alert_system.receiveAlert(zone3)

# Process all received fire alerts
alert_system.processAlert()
alert_system.processAlert()
alert_system.processAlert()

# Emergency vehicles that will be dispatched to the fire zones
vehicle1 = EmergencyVehicle("V1", "Fire Truck")
vehicle2 = EmergencyVehicle("V2", "Helicopter")
vehicle3 = EmergencyVehicle("V3", "Fire Truck")

# Emergency response system to manage the vehicles
response_system = EmergencyResponse()
response_system.addVehicle(vehicle1)
response_system.addVehicle(vehicle2)
response_system.addVehicle(vehicle3)

# Dispatch emergency vehicles to fire zones
response_system.dispatchVehicle()
response_system.dispatchVehicle()

# Evacuation system to manage people needing evacuation
evacuation_system = EvacuationQueue()
evacuation_system.addEvacuee("John ")
evacuation_system.addEvacuee("Smith")
evacuation_system.addEvacuee("Brown")

# Evacuate people in the queue
evacuation_system.startEvacuation()
evacuation_system.startEvacuation()
evacuation_system.startEvacuation()
```

Test Results:

- Fire alerts were processed correctly.
- Emergency vehicles were dispatched properly.
- All evacuees were evacuated without any crash.
- Example print outputs showed successful system operations.

6. Showcasing the Achievement of Project Goals (10%)

The project met its goals by:

- Simulating wildfire alerts and emergency responses.
- Using simple OOP structures like classes and methods.
- Using Queues and Deques to simulate real-world order of operations.

Example output:

```
➡ New alert received: Zone 101: Intensity 8, Location Forest area A
New alert received: Zone 102: Intensity 5, Location Forest area B
New alert received: Zone 103: Intensity 9, Location Mountain area C
Processing alert for Zone 101: Intensity 8, Location Forest area A
Processing alert for Zone 102: Intensity 5, Location Forest area B
Processing alert for Zone 103: Intensity 9, Location Mountain area C
Added vehicle Fire Truck - V1 to the emergency response queue.
Added vehicle Helicopter - V2 to the emergency response queue.
Added vehicle Fire Truck - V3 to the emergency response queue.
Dispatched vehicle Fire Truck - V1 to the emergency site.
Dispatched vehicle Helicopter - V2 to the emergency site.
Evacuee John added to the evacuation queue.
Evacuee Smith added to the evacuation queue.
Evacuee Brown added to the evacuation queue.
Evacuating John .
Evacuating Smith.
Evacuating Brown.
```

7. Discussion and Conclusions (10%)

Through this project, we applied core OOP concepts like encapsulation (keeping the queue/deque lists private inside classes).

I also understood better why using structures like queues is important; they automatically

manage correct ordering without manual tracking.

Limitations: The system is simple and does not handle real-time changes or priorities.

Future improvements could include adding fire severity-based prioritization or multi-threading for faster simulation.