

## IMPLEMENTATION NOTES FOR LAB EXERCISE 6

KYEREMANTENG, PRINCE SAMUEL

2 2 2 5 6 5 2 7

---

### Lab 6: Fibonacci Heaps (Improving Traffic Flow in Urban Areas)

This code implementation uses the Fibonacci Heaps to Improve Traffic Flow in Urban Areas. For the purpose of the assignment, we are considering University of Ghana Campus as the Urban area.

#### Pseudocode Implementation

CLASS FibNode:

PROPERTIES:

- intersection\_id: string
- traffic\_volume: float
- degree: integer
- marked: boolean
- parent, child, left, right: FibNode references

CLASS FibonacciHeap:

PROPERTIES:

- min\_node: FibNode
- total\_nodes: integer

FUNCTION insert (intersection\_id, traffic\_volume):

- Create new FibNode
- IF heap empty:
  - Set as min\_node
- ELSE:
  - Add to root list
  - Update min\_node if necessary
- Increment total\_nodes
- Return new node

FUNCTION extract\_min():

- IF min\_node exists:
  - Move children to root list
  - Remove min\_node from root list
  - Consolidate heap
  - Decrement total\_nodes
- Return extracted node

FUNCTION consolidate ():

- Create degree array
- Group nodes by degree
- Rebuild heap structure
- Update min\_node

## IMPLEMENTATION NOTES FOR LAB EXERCISE 6

KYEREMANTENG, PRINCE SAMUEL

2 2 2 5 6 5 2 7

---

CLASS TrafficManagementSystem:

PROPERTIES:

heap: FibonacciHeap

intersection\_nodes: Dictionary

FUNCTION add\_intersection(id, volume):

Insert into heap

Store node reference in dictionary

FUNCTION get\_critical\_intersection():

Extract minimum from heap

Return intersection ID

FUNCTION optimize\_traffic\_signals():

WHILE heap not empty:

Get critical intersection

Perform signal optimization

### System Constraints

#### 1. Traffic Volume Metrics

- Range: 1.0 to 5.0
- Lower values indicate higher congestion
- Floating-point representation

#### 2. Intersection Properties

- Unique string identifiers
- Format: [Location]-[Type]-[Number]
- No duplicate intersection IDs allowed

#### 3. Memory Management

- All data stored in memory
- No persistent storage implemented
- No data recovery mechanisms

#### 4. Heap Properties

- Minimum-heap implementation
- Amortized  $O(1)$  for insert
- $O(\log n)$  for extract-min
- No maximum size limit

## IMPLEMENTATION NOTES FOR LAB EXERCISE 6

KYEREMANTENG, PRINCE SAMUEL

2 2 2 5 6 5 2 7

---

### Limitations

1. Real-time Constraints
  - No real-time traffic data integration
  - Simulated traffic volumes only
  - No dynamic updates during operation
2. Optimization Scope
  - Optimization is done for single insertion only
  - The algorithm suffers when the intersections are  $\geq 10$
  - There is no coordination between adjacent intersections
  - No consideration of traffic patterns or time-of-day
3. Scalability
  - Single intersection optimization only
  - No coordination between adjacent intersections
  - No consideration of traffic patterns or time-of-day