FlowRates

- northRoadFlow: Dict{east: int. south: int. west: int}
- eastRoadFlow: Dict{north: int, south: int, west: int}
- southRoadFlow: Dict{north: int, east: int, west: int}
- westRoadFlow: Dict{north: int, east: int, south: int}
- driveOnLeft: boolean
- + getDriveOnLeft(): boolean
- + getNorthRoadFlow(east/south/west): int
- + getEastRoadFlow(north/south/west): int
- + getSouthRoadFlow(north/east/west): int
- + getWestRoadFlow(north/east/south): int

Junction

- northLanes: Direction
- eastLanes: Direction
- southLanes: Direction
- westLanes: Direction
- configuration: Parameters
- flowRates: FlowRates
- + setFlowRates(FlowRates): void
- + setJunctionConfiguration(Parameters): void
- + runSimulation(): ResultSet

Direction

- pools: Dict{north: int, east: int, south: int, west: int}
- lanes: Lane[]
- dedicated ane. Fnum
- dedicatedLaneFlow: int
- maxLength: int - maxWait: float
- avgWait: float
- + simulateUpdate(): void
- + getMaxLength(): int
- + getMaxWait(): float
- + getAvgWait(): float

ResultSet

- northRoad: Dict{maxLength: int. maxWait: float. avgWait: float} - eastRoad: Dict{maxLength: int, maxWait: float, avgWait: float}
- southRoad: Dict{maxLength: int, maxWait: float, avgWait: float}
- westRoad: Dict{maxLength: int, maxWait: float, avgWait: float}
- overallScore float
- + aetNorth(): Dict
- + getEast(): Dict
- + getSouth(): Dict
- + aetWest(): Dict
- + getScore(): float

Parameters

- noLanes: int∏ - dedicatedLane: Enum[]
- dedicatedLaneFlow: int[] - pedestrianCrossing: boolean[]
- crossingTime: int/float[]
- crossingRPH: int/float[]
- sequencingPriority: int[]
- + getNoLanes(): int[]
- + getDedicatedLane(): Enum[] + getDedicatedLaneFlow(): int[]
- + hasPedestrianCrossing(): boolean[]
- + getCrossingTime(): int/float[]
- + getCrossingRPH: int/float[]
- + getSequencingPriority: int∏

Lane

- currentVehicles: Queue
- queueLimit: int
- flowing: boolean
- directionTo: Direction[]
- directionFrom: Direction
- + addVehicles(int): void
- + setFlowing(boolean): void



Vehicle

- directionFrom: Enum - directionTo: Enum
- + getDirFrom(): Enum
- + getDirTo(): Enum