# Capacitated Vehicle Routing Problem

## 1.1 Problem Statement

The Capacitated Vehicle Routing Problem (CVRP) is a classical combinatorial optimization problem in the field of logistics and transportation. It involves determining the optimal set of routes for a fleet of vehicles that must deliver goods to a set of customers from a central depot. Each customer has a specific demand, and each vehicle has a limited carrying capacity.  
The goal is to minimize the total distance traveled (or total cost), while ensuring that:

* Each customer is visited exactly once by a single vehicle
* The total demand on any route does not exceed the vehicle’s capacity
* All routes start and end at the central depot.

This problem is critical in real-world logistics and supply chain operations, such as delivery services, waste collection, and distribution of goods, where efficient route planning can significantly reduce operational costs and environmental impact.  
In this project, we implement and analyze solution strategies for the Capacitated Vehicle Routing Problem (CVRP), with a particular focus on comparing the performance of evolutionary algorithms—such as Genetic Algorithms—with non-evolutionary and alternative metaheuristic approaches, including Greedy Search, Random Search, and Simulated Annealing.

## 1.2 Background & Importance

The CVRP extends the classical Traveling Salesman Problem by introducing multiple vehicles and capacity constraints, making it significantly more complex and computationally challenging. Since finding exact solutions becomes infeasible for large instances, metaheuristic algorithms such as Genetic Algorithms (GAs) are widely used to obtain near-optimal solutions efficiently.  
  
Solving CVRP effectively has practical importance across various industries, enabling cost reduction, improved delivery performance, and better resource utilization.

## 1.3 Problem Representation

In the context of evolutionary algorithms, representing a solution to the CVRP is a critical design choice that impacts the performance and correctness of the algorithm.  
  
A typical representation is a permutation of customer nodes, where the depot is implied and used to segment routes based on vehicle capacity constraints.

For example, a chromosome might look like:  
*[3, 5, 1, 2, 4, 6]*  
  
This sequence represents the order in which customers are visited. To convert this into a valid CVRP solution:

* Start from the depot.
* Traverse the customer list in order.
* Keep track of the vehicle's remaining capacity.
* When the next customer's demand cannot be fulfilled, insert a depot return and start a new route with a fresh vehicle.

Example route segmentation with capacity checks:  
*[Depot → 3 → 5 → Depot → 1 → 2 → 4 → Depot → 6 → Depot]*  
  
This structure ensures:

* Each customer is visited exactly once.
* Vehicle capacity is respected.
* All routes start and end at the depot.

## 1.4 Fitness Function

The fitness function is a critical component of the evolutionary algorithm, as it evaluates the quality of each solution (individual) in the population. In the Capacitated Vehicle Routing Problem (CVRP), the fitness of a solution is typically measured by the total distance traveled across all routes.  
  
Objective: Minimize the total distance traveled by all vehicles while fulfilling customer demands and respecting vehicle capacity constraints.  
  
Fitness Calculation Steps:

1. Segment the customer sequence into feasible routes based on vehicle capacity.
2. For each route:  
    - Start and end at the depot.  
    - Sum the distances between consecutive locations (using the distance matrix).
3. Accumulate the total distance across all routes.

Mathematical Formulation:  
Let R = {r₁, r₂, ..., rₖ} be the set of routes,  
Each rᵢ be a sequence of customer visits, starting and ending at the depot,  
d(i, j) be the distance between node i and node j,  
  
Then the total fitness (cost) is:  
f(x) = ∑(r ∈ R) ∑(i=0 to |r|-1) d(rᵢ, rᵢ₊₁),  
Where r₀ and r\_{|r|+1} are both the depot.  
  
Important Notes:

* Lower fitness values indicate better solutions.
* Infeasible solutions (e.g., exceeding capacity) may be penalized or repaired.
* This function guides the selection and evolution process.