# Drone Route Planning

Organization : Indian Space Research Organisation (**ISRO**)
Category : Software
Domain : Security & Surveillance
Problem Statement Number : NM383


**Team Name:** TeamALPHA

**Team Leader :** PABBISETTY SRI RANGA

**Team Members:**
POOJARI VENKATESH
PUNEETH YASHASVI KASHYAP APPARASU
KURUGUNDLA LAKSHMI DEEPTHI
VISHNU VARDHAN GOTTUMUKKALA
YASWANTH REDDY BYTASANDRAM


SIH

SMART INDIA HACKATHON 2020

# Problem Description:

Develop an application for automatically planning a route (for shortest time to cover area) and schedule of drones for mapping a given area. The input provided will be.

1) Map of area to be covered. (Shapefile)
2) Number of drones.
3) Range (in km based on battery life)
4) Top speed
5) Location of automatic charging station The software must have features for visualizing a simulated animation of the plan.

## IDEA DESCRIPTION:

Our drone route planning system uses PyShp library to read the shape file and outputs polygon by using matplotlib we plot the points all across the graph which are distanced by √2 times the radius of the drone coverage.

By using K-means Clustering we make n-clusters from points (n = number of drones). Now drones take in their course such that each cluster corresponds to a particular drone.

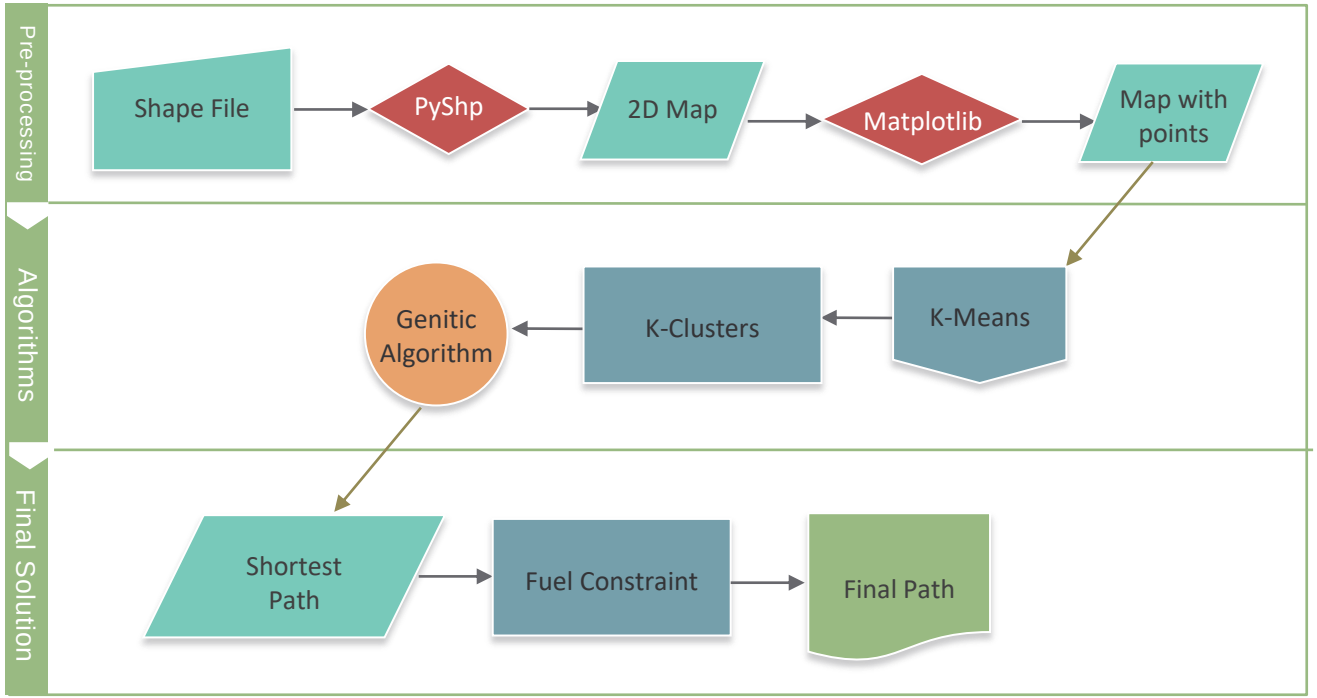
Travelling Salesman Problem will map the shortest possible route a drone can take to cover all the points in it's corresponding cluster. The drone would have an initial capacity which would be decreased due to weight of the path taken by the drone. A condition that the capacity of the drone must never reach negative is applied.

If the drone finds it's capacity to be reaching a negative value by taking a path, it immediately searches for the nearest recharge station and get recharged to continue the current course.We run this multiple times to ensure optimality. This Travelling Salesman problem can be solved optimally using the Genetic Algorithm.

### TECHNOLOGY STACK:

- Python
- Libraries
  - NumPy
  - Matplotlib
  - Sci-Kit Learn
  - PyShp(For ShapeFile)
- SimPY(Simulations)

# Work Flow


Pre-processing
Shape File
PyShp
2D Map
Matplotlib
Map with points
Algorithms
K-Means
K-Clusters
Genitic Algorithm
Final Solution
Shortest Path
Fuel Constraint
Final Path