# AC21007 – Algorithms and Artificial Intelligence

# Team members:

Gerda Ugne Pupelyte **190013631**

Laura Naslenaite **190013642**

Rael Watt **190011819**

Sem Jasaitytė **190014736**Matthew Bain **190012264**

# Word count: *<822 words – 750 words +/- 10%>*

# 

# AI Project Proposal: *Searching for the closest restaurant of a specific cuisine within the chosen area*

# Summary of Proposal

# The project focuses on identifying an eatery that specialises in a particular cuisine in the vicinity of the chosen location. It can be done by specifying the preferred type of cuisine (goal). Each eatery has a place on a map with a specific longitude and latitude. The map is used to judge the distance between the restaurants and is used to determine the available travelling directions.

# The Problem

The problem is to determine the closest restaurant with a preferred cuisine in the chosen location.

Each node would hold attributes including coordinates of a node (i.e. restaurant), its name and a unique ID along with the state, parent, action and path cost.

* **States:** The state is determined by the cuisine the restaurant provides. The cuisines are determined by the data set. There are *n* restaurants, each with *x* cuisines, where *x* is a positive number. Each node (restaurant) has a location, specified by longitude and latitude. The location is processed using a coordinate system. The agent starts in a restaurant that provides Italian cuisine and wants to find a restaurant with Mexican cuisine.
* **Initial State:** The agent starts in a restaurant that provides Italian cuisine. The restaurant location will be determined by choosing a specific restaurant from the dataset.
* **Actions:**
  + **Scan nearby area:** The eligibility for travelling to a restaurant is determined by the agent’s location. By using a coordinate system, the map of restaurants is examined, and restaurants within the range of the agent for a set distance are shown as available paths to travel to.
  + **Restaurant inspection:** The agent can inspect a restaurant’s cuisine within a set distance (5KM - subject to change) from their current location by travelling to the chosen restaurant. After the travel, the agent’s location is updated to match the visited restaurant’s location.
* **Transition model:** the state is changed by the agent going from the current restaurant(i.e. node) to another one that is in the searching distance, and inspecting its cuisine. The state is then updated with the latter restaurant’s cuisine. The agent can then continue travelling to another restaurant if the goal is not met.
* **Goal test:** Checks whether the current restaurant’s state matches the preferred cuisine of the Agent (Mexican).
* **Path cost:** Each step costs the distance between the restaurants, so the path cost is the distance between the starting location and the final restaurant.

# 

# Why it Matters

# The problem offers an efficient way to find a close restaurant with preferable cuisine. This is based on your original location, saving time by not having to explore every single option there is. An optimal solution is provided by saving the agent’s time - considering the path cost (distance) to the destination.

# The Data

# The data needed is a list of available restaurants, their location (provided by longitude and latitude) and the provided cuisines. The data could be mined online, or made by the team.

A proposed data set is Zomato Restaurants data [1]. The needed data could be filtered from the JSON files (title, cuisines and longitude/latitude). The location data could be included as part of the node. A map could be set up by consulting the given data, to which the agent could refer to in order to see the closest restaurants available for inspection.

# Solutions and Expected Results

Uninformed searches could utilize a depth-limited search and uniform-cost search. A Depth-limited search is not guaranteed to find a solution, and the algorithm does not ensure an optimal solution for problems where more than one goal state could exist. This search could reach the goal (i.e. a restaurant with a preferred cuisine), yet it may not be the closest one to the initial state.

The uniform-cost search would provide an optimal solution regarding the path cost (in this case - distance) with guaranteed completeness. The algorithm would ensure that the goal state could be reached in the shortest distance possible even though there might be other solutions available. Unfortunately, there is a likelihood of infinite exploration.

For the informed search strategies, a heuristic function has to be considered first.

For the design of the heuristic function, we would want to prioritize the shortest road to the goal. This could be achieved by using Best-First (Greedy) search. However, the most straightforward way to the goal is not always the most optimal. This can be tackled by using A\* search, which combines both heuristics and the lowest path cost prioritization to find the most optimal solution.

Regarding the problem's complexity, the branching factor could be expected to be very large while the depth and maximum length of any path could be relatively short if the initial state is chosen to be in the densely populated area and oppositely could occur in the rural district. Time and space complexity would be mainly influenced by the adopted search algorithms.

**References**

[1] Metha, S. 2019. *Zomato Restaurants Data*. [Accessed on 9 March 2021] [Online]. Available from: <https://www.kaggle.com/shrutimehta/zomato-restaurants-data>