

AI Assignment 2

- **Rupanshu Yadav - 2019475**
- python scripts are used to extract the csv and convert it into prolog knowledge-base. The following code converts csv to pandas and then converts to *.pl facts.

```
import pandas as pd

df = pd.read_excel('../roaddistance.xlsx')

df = df.drop('Road Distance of Major Cities', axis=1)
df.columns = df.iloc[0]
df = df.drop(df.index[0])

f = open('../src/knowledgeBase.pl', 'w')

for i in range(len(df)):
    start_city = df[df.columns[0]].iloc[i]
    start_city = start_city.lower()
    for j in df.columns[1:]:
        distance = df[j].iloc[i]
        if distance == '-':
            continue
        end_city = j.lower()
        f.write(f'connect({start_city}, {end_city}, {distance}).\n')
        f.write(f'connect({end_city}, {start_city}, {distance}).\n')

f.close()
```

- For heuristic, the air distance between cities is used. To get this distance distance24 api is used. Following code demonstrates the api use and convert it to a prolog file.

```
import requests
import json

response = requests.get("https://www.distance24.org/route.json?stops=Hamburg|Berlin")
url = 'https://www.distance24.org/route.json?stops='

import pandas as pd

df = pd.read_excel('../roaddistance.xlsx')

df = df.drop('Road Distance of Major Cities', axis=1)
df.columns = df.iloc[0]
df = df.drop(df.index[0])

f = open('../src/Heuristic.pl', 'w')
```

```

for i in range(len(df)):
    start_city = df[df.columns[0]].iloc[i]
    start_city_s = start_city.lower()
    for j in df.columns[1:]:
        end_city_s = j.lower()
        url_end = f'{start_city_s}|{j_s}'
        print(url_end)
        response = requests.get(url + url_end)
        if response.status_code != 200:
            continue
        dic = response.json()
        distance = dic['distance']
        if distance == 0:
            continue
        print(f'connect_air({start_city_s}, {end_city_s}, {distance}).')
        f.write(f'connect_air({start_city_s}, {end_city_s}, {distance}).\n')
        f.write(f'connect_air({end_city_s}, {start_city_s}, {distance}).\n')

f.close()

```

- Working screenshots of the program.

```

Choose search method:
1. Best First Search
2. Depth First Search
|: 2.
Enter the name of start city:
|: delhi.
Enter the name of end city:
|: bombay.
delhi agartala ahmedabad agra bangalore allahabad bhubaneshwar amritsar bombay 15588
true .

?- █

```

Use of Depth First Search

```
Choose search method:
1. Best First Search
2. Depth First Search
|: 1.
Enter the name of start city:
|: delhi.
Enter the name of end city:
|: bombay.
delhi bombay 1404
true .

?- 
```

Use of Best First Search

Steps to run search

- Open prolog in terminal

```
swipl
```

- Enter the command to include prolog files

```
[knowledgeBase, heuristic, search].
```

- Run the input predicate

```
input.
```