

Uninformed and Informed Search Max

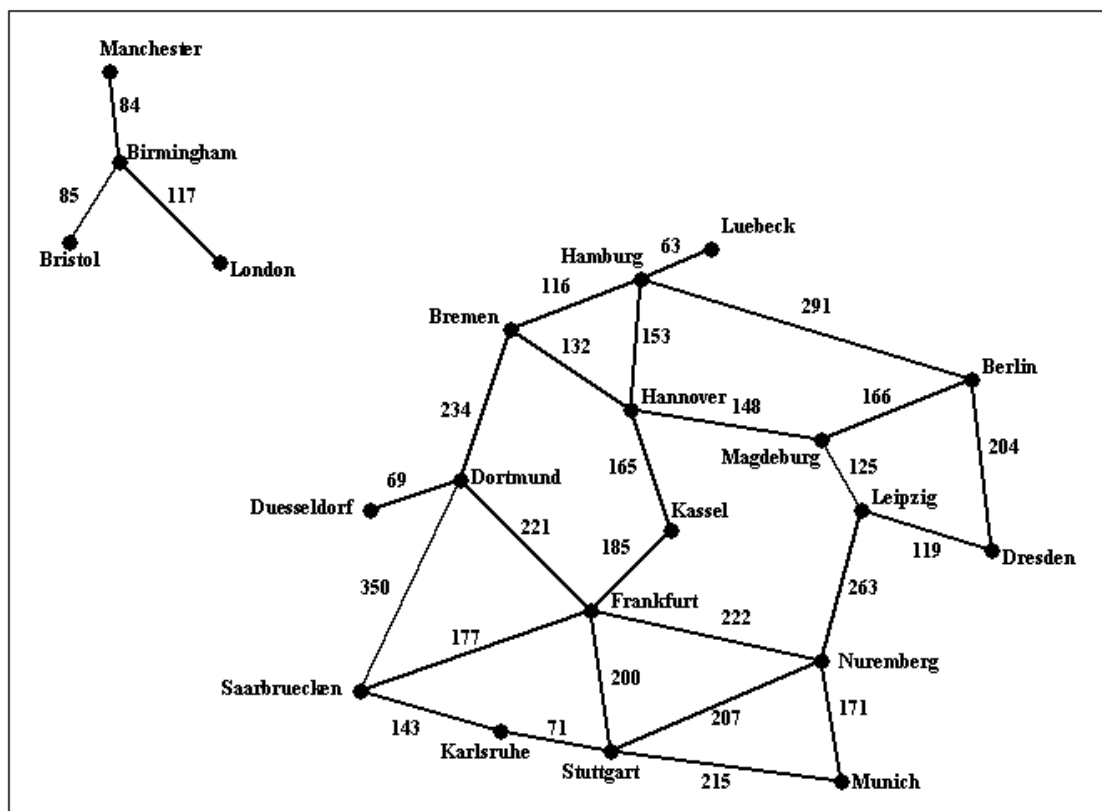


Figure 1: Visual representation of [input1.txt](#)

Implement a search algorithm that can find a route between any two cities. Your program will be called `find_route`, and will take exactly commandline arguments as follows:

find_route input_filename origin_city destination_city heuristic_filename

An example command line is:

`find_route input1.txt Bremen Kassel` (For doing Uninformed search)

or

`find_route input1.txt Bremen Kassel h_kassel.txt` (For doing Informed search)

If heuristic is not provided then program must do uninformed search. Argument `input_filename` is the name of a text file such as [input1.txt](#), that describes road connections between cities in some part of the world. For example, the road system described by file `input1.txt` can be visualized in Figure 1 shown above. You can assume that the input file is formatted in the same way as [input1.txt](#): each line contains three items. The last line contains the items "END OF INPUT", and that is

how the program can detect that it has reached the end of the file. The other lines of the file contain, in this order, a source city, a destination city, and the length in kilometers of the road connecting directly those two cities. Each city name will be a single word (for example, we will use New_York instead of New York), consisting of upper and lowercase letters and possibly underscores.

The program will compute a route between the origin city and the destination city, and will print out both the length of the route and the list of all cities that lie on that route. It should also display the number of nodes expanded, nodes generated and max number of nodes in the fringe. For example,

```
find_route input1.txt Bremen Kassel
```

should have the following output:

```
nodes expanded: 12
nodes generated: 19
max nodes in memory: 11
distance: 297.0 km
route:
Bremen to Hannover, 132.0 km
Hannover to Kassel, 165.0 km
```

and

```
find_route input1.txt London Kassel
```

should have the following output:

```
nodes expanded: 7
nodes generated: 6
max nodes in memory: 3
distance: infinity
route:
none
```