A brute-force solution to problem “tsp”

As noted in the “Five example optimization problems” handout, problem tsp is intrinsically a discrete optimization problem. In fact it is a combinatorial and graph-theoretic optimization problem. Such problems are a topic in MATH 663 Graph Theory. For that reason, and also because tsp fits neither with the continuous optimization problems of the rest of the course, nor the techniques in the textbook, we will not cover it further in MATH 661 Optimization, beyond the brute-force solution tsp.m on this document. There is no claim that this code represents a good approach, but it does solve the problem. Please see a course or book on combinatorial optimization or graph theory for more information on this kind of problem.

The MATLAB/OCTAVE code tsp.m starts with a generally-useful and basic idea, namely that one may represent an edge-weighted graph\(^1\) by a matrix. First, the vertices (cities) A, F, J, N, S, W have indices 1, 2, 3, 4, 5, 6, respectively, in this data structure. Then the matrix \(M = \text{edgew}\) has entry \(m_{ij} = -1\) if there is no flight between city (vertex) \(i\) and city \(j\); otherwise \(m_{ij}\) is the cost of the flight from city \(i\) to city \(j\).

If the seven-city itinerary \(p = i_1 i_2 i_3 i_4 i_5 i_6 i_7\) is feasible, in the sense that the corresponding flights exist, then the cost of \(p\) is the sum of the six flights,

\[
C(p) = m_{i_1 i_2} + m_{i_2 i_3} + m_{i_3 i_4} + m_{i_4 i_5} + m_{i_5 i_6} + m_{i_6 i_7} = \sum_{k=1}^{6} m_{i_k i_{k+1}}.
\]

This cost is computed by a for loop:

```matlab
C = 0; % C = cost of path (or -1)
for k = 1:6
    w = edgew(p(j,k),p(j,k+1));
    if w < 0
        C = -1;
        break
    else
        C = C + w;
    end
end
```

If the itinerary is not feasible we “break” out of the for loop and move to the next case.

The above explains how to find the optimal itinerary among the feasible ones. How to generate all feasible itineraries? The brute force approach here is to pre-generate all itineraries using MATLAB/OCTAVE’s perms function, which generates all permutations of a given vector. Because the itineraries start and end in city S, we generate the permutations of the other cities ([1 2 3 4 6]) and then prepend and append \(S = 5\). Then, in a big for loop, we test whether the itinerary is feasible and compute its cost if so. Note the small

\(^1\)A graph is a collection of vertices with edges which can be defined as pairs of vertices. It is one of the basic objects of (discrete) mathematics.
anonymous function `printcase`, taking three arguments, whose only purpose is to print out an itinerary. Running the code finds only four feasible itineraries and identifies one of the two optimal solutions:

```
>> tsp
feasible SANFWJS : 1250
feasible SFNAWJS : 1200
feasible SJWANFS : 1200
feasible SJWFNAS : 1250
optimal SFNAWJS : 1200
```