

```

% *** Solução:

% A solução será uma lista de viagem/2
% viagem(Vid, L) é um termo que indica que o veículo vid efectua as etapas
% na lista L.
% etapa(0, D, Lcarga) é um termo que indica que as cargas em Lcarga são
% transportadas de 0 para D.
%
% Exemplo:
% [viagem(v1, [etapa(a, p, [c2])]),
%  viagem(v2, [etapa(b, p, [c1]), etapa(p, vr, [c1, c2]), etapa(vr, br, [c2])])
% ]

% planos/2
% plano(L, C) :- L é a melhor lista de viagens e tem custo C.
plano(L, C) :-
consult('mapa.dat'),
% construir a lista das cargas e suas localizações
findall(Cid/Cloc, carga(Cid,Cloc,_,_,_), Lcargas),
% construir a lista dos veículos e suas localizações
findall(Vid/Vloc, (localidade(Vloc,_,Lv), member(Vid, Lv)), Lveículos),
% obter a lista de todas as soluções possíveis e seleccionar a melhor
setof(Ctemp/Ltemp,
      plano possível(Lcargas, Lveículos, Ltemp, Ctemp), [C/L|_]).

% Define o que é uma solução para o problema e seu custo

plano possível(Lcargas, , Lveículos, L,C) :-
plano possível(Lcargas, Lveículos, [], 0, L, C).

% Versão auxiliar com acumuladores.
% plano possível(lista de carga/localização,
%               lista de veículo/localização,
% Acumulador de viagens,
% Acumulador de custo
% Resultado de viagens,
% Resultado de custo)

% Se não há carga, terminamos...
plano possível([], _, AcViagens, AcCusto, AcViagens, AcCusto).

% Se existe carga para transportar, temos as seguintes situações...

% existe veículo sem viagem associada junto à carga e é possível
% criar uma nova viagem com uma etapa que coloca a carga no destino
plano possível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-

```

```

        % seleccionar uma carga e uma ligação
append(Lc1, [Cid/Orig|Lc2], Lcargas),
    carga(Cid, _, Dest, Peso, _),
ligação(Orig, Dest, D, T),

% seleccionar um veículo sem viagem em Orig com capacidade para a carga
...
\+member(viagem(Vid, _), AcViagens),
...

% calcular nova lista de cargas
append(Lc1, Lc2, LcNew),
% calcular nova lista de veículos
...
% calcular novo custo
AcCnew is ...

    planopossível(LcNew, LvNew,
        [viagem(Vid, [etapa(Orig, Dest, [Cid])])|AcViagens], AcCnew, L, C).

% existe veículo sem viagem associada junto à carga e é possível
% criar uma nova viagem com uma etapa que não coloca a carga no destino
planopossível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-
    % seleccionar uma carga e um veículo com capacidade
    ...
% seleccionar uma ligação que não coloque a carga no destino
...
% calcular novos valores
...
    planopossível(...).

% existe veículo com viagem associada junto à carga e é possível
% adicionar uma etapa nessa viagem que coloca a carga no destino
planopossível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-
    ...

% existe veículo com viagem associada junto à carga e é possível
% adicionar uma etapa nessa viagem que não coloca a carga no destino
planopossível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-
    ...

% é possível utilizar uma etapa existente que coloca a carga no destino
planopossível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-
    ...

% é possível utilizar uma etapa existente que não coloca a carga no destino

```

```
planopossível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-  
...  
  
% não existe veículo junto à carga e é possível criar uma nova viagem  
% para um qualquer veículo com uma etapa sem carga  
planopossível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-  
...  
  
% não existe veículo junto à carga e é possível criar uma nova etapa sem carga  
% numa viagem já existente  
planopossível(Lcargas, Lveic, AcViagens, AcCusto, L, C) :-  
...
```