

## HECHOS

```
Program x +
1 %HECHOS
2 % conexión(Ciudad, Destino, Costo).
3
4 conexión(vancouver, edmonton, 16).
5 conexión(vancouver, calgary, 13).
6
7 conexión(edmonton, saskatoon, 12).
8
9 conexión(calgary, regina, 14).
10 conexión(calgary, edmonton, 4).
11
12 conexión(saskatoon, calgary, 9).
13 conexión(saskatoon, winnipeg, 20).
14
15 conexión(regina, winnipeg, 4).
16 conexión(regina, saskatoon, 7).
17
```

## REGLAS

```
existe_conexión(Ciudad, Destino) :-
    conexión(Ciudad, Destino, _).

valor(Ciudad, Destino, Costo) :-
    conexión(Ciudad, Destino, Costo).

% aristas en ambos sentidos
existe_arista(Ciudad, Existe) :-
    conexión(Ciudad, Existe, _).

existe_arista(Ciudad, Existe) :-
    conexión(Existe, Ciudad, _).

% costo para ir de una ciudad inicial a una ciudad final pasando por una ciudad intermedia.
costo_para_ir_desde(Ciudad_i, Ciudad_m, Ciudad_f, CostoTotal) :-
    conexión(Ciudad_i, Ciudad_m, Costo1),
    conexión(Ciudad_m, Ciudad_f, Costo2),
    CostoTotal is Costo1 + Costo2.

% Es posible viajar de Ciudad_i a Ciudad_f pasando por Ciudad_m
es_posible_viajar(Ciudad_i, Ciudad_m, Ciudad_f) :-
    conexión(Ciudad_i, Ciudad_m, _),
    conexión(Ciudad_m, Ciudad_f, _).
```

## DEMOSTRACIONES

1.

```
existe_conexión(Ciudad, Destino) :-  
    conexión(Ciudad, Destino, _).
```

```
⚙ existe_conexión(saskatoon, Destino)  
  
Destino = calgary  
Destino = winnipeg  
  
?- existe_conexión(saskatoon, Destino)
```

2.

```
valor(Ciudad, Destino, Costo) :-  
    conexión(Ciudad, Destino, Costo).
```

```
⚙ valor(calgary, Destino, Costo).  
  
Costo = 14,  
Destino = regina  
Costo = 4,  
Destino = edmonton  
  
?- valor(calgary, Destino, Costo).
```

3.

```
% aristas en ambos sentidos  
existe_arista(Ciudad, Existe) :-  
    conexión(Ciudad, Existe, _).  
  
existe_arista(Ciudad, Existe) :-  
    conexión(Existe, Ciudad, _).
```

```
⚙ existe_arista(calgary, Existe)  
  
Existe = regina  
Existe = edmonton  
Existe = vancouver  
Existe = saskatoon  
  
?- existe_arista(calgary, Existe)
```

4.

```
% costo para ir de una ciudad inicial a una ciudad final pasando por una ciudad intermedi
costo_para_ir_desde(Ciudad_i, Ciudad_m, Ciudad_f, CostoTotal) :-
    conexión(Ciudad_i, Ciudad_m, Costo1),
    conexión(Ciudad_m, Ciudad_f, Costo2),
    CostoTotal is Costo1 + Costo2.
```

```
costo_para_ir_desde(edmonton, saskatoon, winnipeg, Costo).
```

**Costo** = 32

```
?- costo_para_ir_desde(edmonton, saskatoon, winnipeg, Costo).
```

5.

```
% Es posible viajar de Ciudad_i a Ciudad_f pasando por Ciudad_m
es_posible_viajar(Ciudad_i, Ciudad_m, Ciudad_f) :-
    conexión(Ciudad_i, Ciudad_m, _),
    conexión(Ciudad_m, Ciudad_f, _).
```

```
es_posible_viajar(edmonton, _, calgary).
```

true

```
es_posible_viajar(edmonton, T, calgary).
```

**T** = saskatoon

```
?- es_posible_viajar(edmonton, T, calgary).
```

**ProglIIIG101-Act02-**  
**Santiago Henao**  
**-Johany Ballesteros**  
**-Juan Guillermo Galindo**