|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Materia:** |  | Programación Lógica y Funcional | | |
| **Práctica:** |  | Ejercicio: 99 botellas de cerveza en la pared | | |
| **Alumno(s):** |  | Luis Manuel Cárdenas Ibarra | **Fecha:** | 01/04/2025 |

**Objetivo**

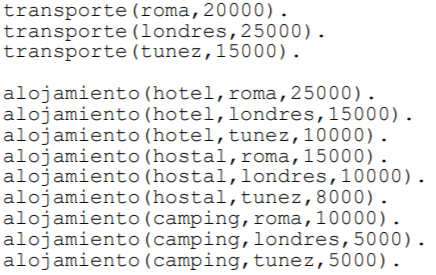
Con este trabajo se se busca realizar 3 ejercicios en Prolog, los cuales son los siguientes:

* **Planificador de gastos**: Calcular el costo total de un viaje considerando transporte y alojamiento.
* **Horóscopo**: Validar fechas de nacimiento y determinar el signo zodiacal correspondiente.
* **Diagnóstico de enfermedades**: Implementar un sistema experto básico para diagnosticar enfermedades y recomendar medicamentos.

Las reglas para los ejercicios ya se nos proporcionaron en el documento adjunto de la práctica.

**Desarrollo**

**Planificador de gastos:**

Primero se definieron las reglas que ya se nos había compartido.

Una vez definidas las reglas, tenemos que hacer la regla principal, la cual se encargará de obtener el costo total del viaje multiplicando el tipo de alojamiento, por los días que se durará el viaje dependiendo del lugar a donde vaya, lo podemos lograr de la siguiente forma.

costo\_viaje(Lugar, TipoAlojamiento, Dias, CostoTotal) :-

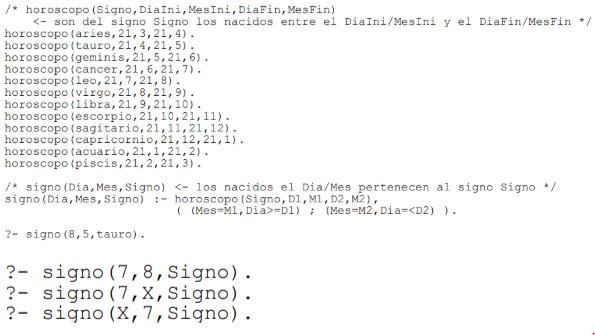
    transporte(Lugar, CostoTransporte),

    alojamiento(TipoAlojamiento, Lugar, CostoPorDia),

    CostoTotal is CostoTransporte + (CostoPorDia \* Dias).

**Horóscopo:**

Para hacer el programa horóscopo, también copiamos las reglas predefinas.



En este caso, lo que tenemos que hacer es modificar el código para validar todo el rango que abarcan los horóscopos, de la siguiente forma podemos agregar cuantos días tiene cada mes y poder saber si está dentro del rango o no.

dias\_mes(1, 31).  % enero

dias\_mes(2, 28).  % febrero

dias\_mes(3, 31).  % marzo

dias\_mes(4, 30).  % abril

dias\_mes(5, 31).  % mayo

dias\_mes(6, 30).  % junio

dias\_mes(7, 31).  % julio

dias\_mes(8, 31).  % agosto

dias\_mes(9, 30).  % septiembre

dias\_mes(10, 31). % octubre

dias\_mes(11, 30). % noviembre

dias\_mes(12, 31). % diciembre

Con esto definido, podremos realizar la regla principal, la cual se va a encargar de hacer las operaciones necesarias .

signo(Dia, Mes, Signo) :-

    dias\_mes(Mes, MaxDias),

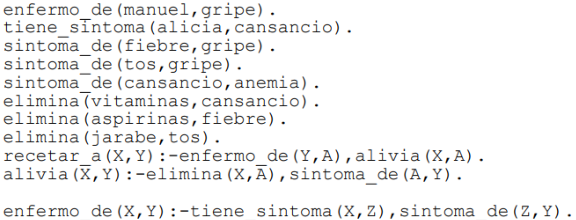
    Dia >= 1, Dia =< MaxDias,

    horoscope(Signo, D1, M1, D2, M2),

    ( (Mes = M1, Dia >= D1) ; (Mes = M2, Dia =< D2) ).

**Diagnóstico de enfermedades:**

Igualmente, lo primero es copiar las reglas brindas.



Con la regla “recetar\_a” podemos sabe que medicamento recetar a cada paciente dependiendo de lo que padezca. Con la regla “alivia” podemos saber que enfermedad se alivia con que medicamento. Por último, con la regla “enfermo\_de” podemos saber que enfermedad tiene cada paciente.

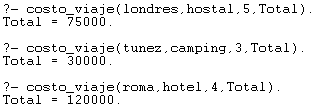
recetar\_a(X,Y):-enfermo\_de(Y,A),alivia(X,A).

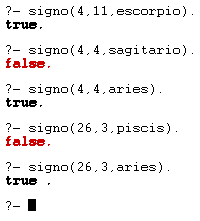
alivia(X,Y):-elimina(X,A),sintoma\_de(A,Y).

enfermo\_de(X,Y):-tiene\_sintoma(X,Z),sintoma\_de(Z,Y).

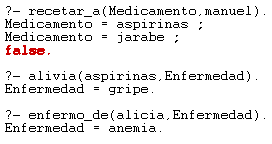
**Resultado**

**Planificador de gastos:**

****

**Horóscopos:**

**Diagnóstico de enfermedad:**



**Conclusiones**

Los ejercicios resolvieron problemas diversos usando **lógica de predicados**, demostrando la versatilidad de Prolog para:

* **Cálculos estructurados** (viajes).
* **Validación de datos** (fechas y horóscopos).
* **Razonamiento automático** (diagnóstico médico).

Este trabajo evidenció cómo Prolog permite implementar soluciones claras y eficientes para problemas que requieren manejo de relaciones lógicas y validación de restricciones, destacando su potencial en aplicaciones de inteligencia artificial y sistemas expertos.