

**UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA
FACULTAD INGENIERÍA EN SISTEMAS CENTRO
UNIVERSITARIO ANTIGUA GUATEMALA**

Inteligencia Artificial



Proyecto 1 Prolog

Manuel Alessandro Concoba Núñez
0910-21-20822

Antigua Guatemala 12 de Abril, 2024

Codigo Fuente

```
% CARRERAS
carrera(ca1, ingenieria_en_sistemas).

% CICLOS
ciclo(ci1, primer_ciclo, s1).
ciclo(ci2, segundo_ciclo, s2).
ciclo(ci3, tercer_ciclo, s1).
ciclo(ci4, cuarto_ciclo, s2).
ciclo(ci5, quinto_ciclo, s1).
ciclo(ci6, sexto_ciclo, s2).
ciclo(ci7, septimo_ciclo, s1).
ciclo(ci8, octavo_ciclo, s2).
ciclo(ci9, noveno_ciclo, s1).
ciclo(ci10, decimo_ciclo, s2).

% CURSOS
% curso(ID, CarreraID, Nombre, CicloID, Prerrequisito)
curso(cu1, ca1, contabilidad, ci1, n).
curso(cu2, ca1, logica_de_sistemas, ci1, n).
curso(cu3, ca1, fisica_i, ci2, n).
curso(cu4, ca1, programacion_i, ci2, n).
curso(cu5, ca1, programacion_ii, ci3, cu4).
curso(cu6, ca1, fisica_ii, ci3, cu3).
curso(cu7, ca1, base_de_datos_i, ci6, n).
curso(cu8, ca1, base_de_datos_ii, ci7, cu7).
curso(cu9, ca1, electodica_digital, ci7, n).

% HORARIOS
% horario(IDHorario, hora(HoraInicio, HoraFin))
horario(h1, hora(7, 9)).
horario(h2, hora(9, 11)).
horario(h3, hora(11, 13)).
horario(h4, hora(14, 16)).
horario(h5, hora(16, 18)).

% Relacionar cursos con horarios (opcional)
curso_horario(cu1, h1).
curso_horario(cu2, h2).
```

```
curso_horario(cu3, h3).
curso_horario(cu4, h4).
curso_horario(cu5, h5).
curso_horario(cu6, h1).
curso_horario(cu7, h2).
curso_horario(cu8, h3).
curso_horario(cu9, h4).
```

```
% ESTUDIANTES
```

```
% estudiante(ID, Nombre, Apellido, Carnet)
estudiante(e1, juan, perez, 12345678).
estudiante(e2, maria, lopez, 23456789).
estudiante(e3, pedro, gomez, 34567890).
estudiante(e4, ana, garcia, 45678901).
estudiante(e5, luis, martinez, 56789012).
```

```
% INSCRIPCIONES
```

```
% inscripcion(CarreraID, EstudianteID)
inscripcion(ca1, e1).
inscripcion(ca1, e2).
inscripcion(ca1, e3).
inscripcion(ca1, e4).
inscripcion(ca1, e5).
```

```
% ASIGNACIONES
```

```
% asignacion_base(EstudianteID, HorarioID, CursoID)
asignacion_base(e1, h1, cu1).
asignacion_base(e1, h2, cu2).
asignacion_base(e1, h3, cu3).
asignacion_base(e2, h1, cu1).
asignacion_base(e2, h4, cu4).
asignacion_base(e3, h5, cu5).
asignacion_base(e4, h5, cu5).
```

```
% Reglas de asignación válidas
```

```
asignacion(E, H, C) :-
    asignacion_base(E, H, C),
    inscripcion(CA, E),
    curso(C, CA, _, _, _),
```

```
horario(H, _).
```

```
% PAGOS
```

```
% pago_base(EstudianteID, CursoID)
```

```
pago_base(e1, cu1).
```

```
pago_base(e1, cu2).
```

```
pago_base(e1, cu3).
```

```
pago_base(e2, cu1).
```

```
pago_base(e2, cu4).
```

```
pago_base(e3, cu5).
```

```
pago_base(e4, cu5).
```

```
% Regla para validar pagos
```

```
pago(E, C) :-
```

```
    pago_base(E, C),
```

```
    inscripcion(CA, E),
```

```
    curso(C, CA, _, _, _).
```

```
% APROBACIONES
```

```
% aprobacion_base(EstudianteID, CursoID)
```

```
aprobacion_base(e1, cu1).
```

```
aprobacion_base(e1, cu2).
```

```
aprobacion_base(e2, cu1).
```

```
aprobacion_base(e2, cu4).
```

```
% Regla para validar aprobaciones
```

```
aprobacion(E, C) :-
```

```
    aprobacion_base(E, C),
```

```
    inscripcion(CA, E),
```

```
    curso(C, CA, _, _, _).
```

```
% CONSULTAS
```

```
% ¿Qué cursos existen en la carrera de ingeniería en sistemas?
```

```
cursos_carrera(CarreraID, Cursos) :-
```

```
    findall(
```

```
        (CursoID, Nombre),
```

```
        curso(CursoID, CarreraID, Nombre, _, _),
```

```
        Cursos
```

).).

% Consulta para obtener en que ciclos esta llevando cursos el estudiante X

```
ciclos_estudiante(EstudianteID, Ciclos) :-  
    findall(  
        (CicloID, CicloNombre, Semestre),  
        (  
            asignacion(EstudianteID, _, CursoID),  
            curso(CursoID, _, _, CicloID, _),  
            ciclo(CicloID, CicloNombre, Semestre)  
        ),  
        Ciclos  
    ).
```

% Consulta para obtener los prerequisitos de un curso

```
prerequisitos_curso(CursoID, Prerrequisitos) :-  
    curso(CursoID, _, _, _, Prerrequisito),  
    ( Prerrequisito = n →  
        Prerrequisitos = []  
    ;  
        findall(  
            (Prerrequisito, Nombre),  
            curso(Prerrequisito, _, Nombre, _, _),  
            Prerrequisitos  
        )  
    ).
```

% Consulta para obtener los estudiantes inscritos en el mismo curso y mismo horario

```
estudiantes_mismo_curso_horario(CursoID, HorarioID, Estudiantes) :-  
    findall(  
        (EstudianteID, Nombre, Apellido, HorarioID, Curso),  
        (  
            asignacion(EstudianteID, HorarioID, CursoID),  
            inscripcion(_, EstudianteID),  
            estudiante(EstudianteID, Nombre, Apellido, _),  
            curso(CursoID, _, Curso, _, _)  
        )  
    ).
```

```
    ),  
    Estudiantes  
).
```

```
% Consulta para obtener los cursos de un estudiante  
cursos_estudiante(EstudianteID, Cursos) :-
```

```
    findall(  
        CursoID,  
        asignacion(EstudianteID, _, CursoID),  
        Cursos  
    ).
```

```
% Consulta para obtener los cursos aprobados por un estudiante  
cursos_aprobados_estudiante(EstudianteID, CursosAprobados) :-
```

```
    findall(  
        CursoID,  
        (  
            aprobacion(EstudianteID, CursoID),  
            inscripcion(_, EstudianteID)  
        ),  
        CursosAprobados  
    ).
```

```
% Consulta para obtener los cursos pendientes de pago un  
estudiante
```

```
cursos_pendientes_pago_estudiante(EstudianteID,  
CursosPendientes) :-  
    findall(  
        CursoID,  
        (  
            asignacion(EstudianteID, _, CursoID),  
            \+ pago(EstudianteID, CursoID)  
        ),  
        CursosPendientes  
    ).
```

Consultas

- ¿Qué cursos existen en la carrera de ingeniería en sistemas?

Codigo consulta

```
courses_carrera(CarreraID, Cursos) :-  
    findall(  
        (CursoID, Nombre),  
        curso(CursoID, CarreraID, Nombre, _, _),  
        Cursos  
    ).
```

Prueba

```
?- courses_carrera(ca1, Cursos).  
Cursos = [(cu1, contabilidad), (cu2, logica_de_sistemas), (cu3, fisica_i), (cu4, programacion_i), (cu5,  
programacion_ii), (cu6, fisica_ii), (cu7, base_de_datos_i), (cu8, base_de_datos_ii), (...)].
```

- ¿En que ciclos esta llevando cursos el estudiante X?

Codigo Consulta

```
ciclos_estudiante(EstudianteID, Ciclos) :-  
    findall(  
        (CicloID, CicloNombre, Semestre),  
        (  
            asignacion(EstudianteID, _, CursoID),  
            curso(CursoID, _, _, CicloID, _),  
            ciclo(CicloID, CicloNombre, Semestre)  
        ),  
        Ciclos  
    ).
```

Prueba

```
?- ciclos_estudiante(e1, Ciclos).  
Ciclos = [(ci1, primer_ciclo, s1), (ci1, primer_ciclo, s1), (ci2, segundo_ciclo, s2)].  
?-
```

- ¿Cuáles son los prerrequisitos del curso Y?

Codigo Consulta

```
prerequisitos_curso(CursoID, Prerrequisitos) :-
    curso(CursoID, _, _, _, Prerrequisito),
    ( Prerrequisito = n →
        Prerrequisitos = []
    ;
        findall(
            (Prerrequisito, Nombre),
            curso(Prerrequisito, _, Nombre, _, _),
            Prerrequisitos
        )
    ).
```

Prueba

```
?- prerequisites_curso(cu5, P).  
P = [(cu4, programacion_i)].
```

- Listar los estudiantes que tienen asignados cursos en el mismo horario.

Codigo Consulta

```
estudiantes_mismo_curso_horario(CursoID, HorarioID,
Estudiantes) :-
    findall(
        (EstudianteID, Nombre, Apellido, HorarioID,
Curso),
        (
            asignacion(EstudianteID, HorarioID,
CursoID),
            inscripcion(_, EstudianteID),
            estudiante(EstudianteID, Nombre, Apellido,
_),

```



```
        curso(CursoID, _, Curso, _, _)  
    ),  
    Estudiantes  
).  
  
Prueba
```

```
?- estudiantes_mismo_curso_horario(C, h5, Es).  
Es = [(e3, pedro, gomez, h5, programacion_ii), (e4, ana, garcia, h5, programacion_ii)].  
?-
```