

Actividad en Parejas – Prolog y Casos Recursivos

En un pequeño pueblo llamado **Villa Lógica**, viven **Ana, Juan, María, Pedro y Sofía**. Cada uno realiza distintas actividades: *cocinar, estudiar, trabajar, jugar* y *descansar*. Además, mantienen **relaciones familiares y de amistad**: Ana es hermana de Pedro; María es prima de Sofía; Juan es amigo cercano de Ana; Pedro y Sofía trabajan juntos; María suele estudiar con Juan en la biblioteca.

En Villa Lógica existen **rutinas**: Ana cocina todos los días salvo cuando está cansada; Juan estudia entre semana y descansa los fines de semana; Pedro trabaja de lunes a viernes y juega al fútbol los sábados; Sofía trabaja en la tienda y estudia idiomas por las noches; María estudia en la universidad y descansa los domingos.

Se pide construir una **base de conocimiento en Prolog** para representar actividades, relaciones y rutinas, y poder responder consultas como: “¿Quién cocina?”, “¿Quiénes son amigos directos e indirectos?”, “¿Quién es familia de quién?”, “¿Qué hace alguien en un día dado?”, “¿Está una actividad en una lista?” o “¿Cuál es la inversa de una lista de actividades?”

Aplicar los conceptos de **caso base** y **caso general** en Prolog mediante hechos, reglas (incluyendo recursión) y pruebas de consultas.

Parte 1 – Hechos básicos

- Define hechos de actividades. Ej.: `cocina(ana).`, `estudia(maria).`, `trabaja(juan).`
- Define hechos de amistad. Ej.: `amigo(ana, juan).`, `amigo(juan, maria).`
- Define hechos familiares (`hermano/prim@`).
- Define hechos de rutinas semanales (días y actividad).

Parte 2 – Reglas simples

- Regla `actividad(X)` que sea verdadera si X realiza alguna actividad (`cocinar/estudiar/trabajar/jugar`).
- Regla `descansa(X)` que sea verdadera si X no realiza otra actividad ese día.
- Regla para verificar parentesco directo (por ejemplo, hermanos o primos).

Parte 3 – Reglas recursivas

- Amistad: *caso base* amistad directa; *caso general* amistad indirecta mediante una tercera persona.
- Miembro en lista (como visto en clase): *caso base* cabeza; *caso general* cola.
- Rutinas por día: regla que determine la actividad de alguien en un día dado.

Parte 4 – Consultas mínimas a mostrar

- `?- cocina(ana).`
- `?- es_amigo(ana, maria).`
- `?- miembro(estudiar, [cocinar, estudiar, jugar]).`
- `?- invertir([cocinar, estudiar, jugar], R).`
- `?- actividad(pedro, sabado, A).`

Parte 5 – Extensión creativa

- Agrega un nuevo personaje con actividades y relaciones.
- Crea al menos una regla nueva que use recursión o combine varias condiciones.
- Incluye al menos dos consultas nuevas.

Parte 6 – Explicación técnica

- En comentarios, explica el **caso base** y el **caso general** de cada regla recursiva.
- Incluye un breve resumen (3–5 líneas) de cómo modelaste las rutinas por día.

Entrega

- Un archivo **.pl** con hechos, reglas, consultas y comentarios explicativos.
- Evidencias de ejecución (capturas o transcripciones) de las consultas mínimas.

Espacios para registrar resultados de consultas:

Consulta: <code>?- cocina(ana).</code> Resultado:

--

Consulta: ?- es_amigo(ana, maria). Resultado:

Consulta: ?- miembro(estudiar, [cocinar, estudiar, jugar]). Resultado:

Consulta: ?- invertir([cocinar, estudiar, jugar], R). Resultado:

Consulta: ?- actividad(pedro, sabado, A). Resultado:

Rúbrica de evaluación

- **Puntualidad:** Entregado a tiempo.
- **Claridad del código:** Legible, organizado y con comentarios útiles.
- **Caso base / general:** Correctos y explicados en comentarios.
- **Pruebas de consultas:** Todas las consultas mínimas ejecutadas y evidenciadas.
- **Creatividad:** Extensión con personaje/reglas/consultas nuevas bien integradas.

Sugerencia de calificación: cada criterio en escala 0–10; promedio final.