

### 4.1 Backtracking

En este capítulo examinaremos en detalle cómo Prolog busca todas las soluciones a un objetivo.

***/\* Programa 4.1 Backtracking \*/***

#### **predicates**

```
hombre(symbol)
mujer(symbol)
padres(symbol,symbol,symbol)
hermana_de(symbol,symbol)
```

#### **clauses**

```
hombre(federico).
hombre(juan).
mujer(alicia).
mujer(victoria).
padres(juan, alicia, federico).
padres(victoria, alicia, federico).
hermana_de(X,Y) :-
    mujer(X),
    padres(X,Madre,Padre),
    padres(Y,Madre,Padre).
```

---

**goal:** hermana\_de (victoria,juan)

## Backtracking

---

Prolog procede de la siguiente manera

- 

- 

- 

- 

Suponer ahora que queremos saber si Victoria es la hermana de alguien. La pregunta escrita en Prolog es

**goal:** hermana\_de(victoria,X)

Para contestar a esta pregunta, Prolog procede de la siguiente manera

**Ejercicio 4.1** Se quiere organizar un torneo de tenis entre los niños de 9 años en un club de tenis. Se pretende encontrar todas las posibles parejas de jugadores de 9 años.

**/\* Programa 4.2 Un torneo de tenis \*/**

### domains

niño = symbol

edad = integer

### predicates

jugador(niño,edad)

### clauses

jugador(pedro,9).

jugador(pablo,10).

jugador(cristina,9).

jugador(susana,9).

---

### goal:

Escribe la pregunta adecuada e indica paso a paso cómo Prolog busca todas las soluciones.

### **/\* Ejercicio 4.3 \*/**

#### **clauses**

a(X,Y):-  
    b(X),  
    c(Y).

a(X,Y):-  
    d(X),  
    b(Y).

b(1).

b(2).

c(1).

c(2).

d(3).

---

**goal** a(3,1)

**goal** a(X,Y)

**/\* Ejercicio 4.4 \*/**

**clauses**

a(X):-  
    b(X,Y),  
    c(Y).

a(X):-  
    c(X).

b(1,2).

b(2,2).

b(3,3).

b(3,4)

c(2).

c(5).

---

**goal** a(1)

**goal** a(2)

**goal** a(3)

**goal** a(4)

### **/\* Ejercicio 4.5 \*/**

#### **clauses**

a(X,Y):-  
    b(X,Y).

a(X,Y):-  
    c(X,Z),  
    a(Z,Y).

b(1,2).

b(2,3).

c(1,2).

c(1,4)

c(2,4).

c(3,4).

---

**goal** a(1,3)

**goal** a(1,X)

**goal** a(2,X)

**goal** a(X,4)

### ***/\* Ejercicio 4.6 \*/***

#### **clauses**

hola(3).

hola(6).

hola(X):-

    Y=X-1,

    hola(Y),

    write(Y).

---

**goal** hola(4)

**goal** hola(9)

**goal** hola(2+1)