

MANIPULACIÓN DE BASES DE DATOS Y SOLUCIONES DE RECOLECCIÓN



MANIPULACIÓN DE BASES DE DATOS (CON SWI – PROLOG INSTALADO)

- PROLOG TIENE 5 COMANDOS BÁSICOS DE MANIPULACIÓN DE BASES DE DATOS:
- SUMAN INFORMACIÓN
 - ASSERT
 - ASSERTA
 - ASSERTZ
- REMUEVEN INFORMACIÓN
 - RETRACT
 - RETRACTALL

EMPEZAMOS

EN PROLOG

- CON LA BASE VACÍA
 - ?- LISTING.
- INCORPORAMOS DATOS
 - ?- ASSERT(HAPPY(PAO)).
 - ?- LISTING(HAPPY).
 - ?- ASSERT(HAPPY(PABLO)),
ASSERT(HAPPY(ALUMNOS_PROGIII)).
 - ?- LISTING(HAPPY).

EN LA BASE

- _____
- HAPPY(PAO).
- HAPPY(PABLO).
- HAPPY(ALUMNOS_PROGIII).

CAMBIANDO EL SIGNIFICADO DE LOS PREDICADOS

- LAS MANIPULACIONES DE LA BASE DE DATOS HAN CAMBIADO EL SIGNIFICADO DEL PREDICADO HAPPY.
- DE MANERA MÁS GENERAL: LOS COMANDOS DE MANIPULACIÓN DE BASES DE DATOS NOS DAN LA CAPACIDAD DE CAMBIAR EL SIGNIFICADO DE LOS PREDICADOS DURANTE EL TIEMPO DE EJECUCIÓN

PREDICADOS ESTÁTICOS Y DINÁMICOS

- LOS PREDICADOS CUYO SIGNIFICADO CAMBIA DURANTE EL TIEMPO DE EJECUCIÓN SE DENOMINAN PREDICADOS DINÁMICOS.
- HAPPY ES UN PREDICADO DINÁMICO
- LOS PREDICADOS ORDINARIOS A VECES SE DENOMINAN PREDICADOS ESTÁTICOS

HACER VALER REGLAS

- ASSERT((NAIVE(X):- HAPPY(X)).
- LISTING.
- HAPPY(PAO).
- HAPPY(PABLO).
- HAPPY(ALUMNOS_PROGIII).
- NAIVE(X):- HAPPY(X).

ELIMINANDO INFORMACIÓN

- AHORA SABEMOS CÓMO AGREGAR INFORMACIÓN A LA BASE DE DATOS PROLOG .
- HACEMOS ESTO CON EL PREDICADO ASSERT.
- ¿CÓMO ELIMINAMOS INFORMACIÓN?.
- HACEMOS ESTO CON EL PREDICADO RETRACT.
- ESTO ELIMINARÁ UNA CLÁUSULA.
- PODEMOS ELIMINAR VARIAS CLÁUSULAS SIMULTÁNEAMENTE CON EL PREDICADO RETRACTALL.

ELIMINANDO CLAUSULAS

- RETRACT(HAPPY(PABLO)).
 - LISTING (HAPPY).
 - RETRACTALL(HAPPY(X)).
 - LISTING (HAPPY).
- HAPPY(PAO).
 - HAPPY(ALUMNOS_PROGIII).
 - NAIVE(X):- HAPPY(X).
 - NAIVE(X):- HAPPY(X).

USANDO ASSERTA Y ASSERTZ

- SI QUEREMOS MÁS CONTROL SOBRE EL ASSERT PODEMOS USAR LAS DOS VARIANTES DE
 - ASSERTA
 - AGREGA AL PRINCIPIO DE LA BASE DE DATOS
 - ASSERTZ
 - AGREGA AL FINAL DE LA BASE DE DATOS

MEMORIZACIÓN

- LA MANIPULACIÓN DE BASES DE DATOS ES UNA ÚTIL TÉCNICA ESPECIALMENTE PARA ALMACENAR LOS RESULTADOS EN CÁLCULOS, EN CASO DE SEA NECESARIO VOLVER A CALCULAR LA MISMA CONSULTA.
- ESTO A MENUDO SE LLAMA MEMORIZACIÓN O ALMACENAMIENTO EN CACHÉ

ELIMINANDO CLAUSULAS

:- DYNAMIC LOOKUP/3.

ADDANDSQUARE(X,Y,RES):-

LOOKUP(X,Y,RES), !.

ADDANDSQUARE(X,Y,RES):-

RES IS (X+Y) * (X+Y),

ASSERT(LOOKUP(X,Y,RES)).

- ADDANDSQUARE(3,7,X).
- ADDANDSQUARE(3,4,X).
- LISTING(LOOKUP).
- RETRACTALL(LOOKUP(_, _, _)).

CORTES ROJOS Y VERDES

ROJOS

:- DYNAMIC LOOKUP/3.

ADDANDSQUARE(X,Y,RES):-

LOOKUP(X,Y,RES), !.

ADDANDSQUARE(X,Y,RES):-

RES IS (X+Y) * (X+Y),

ASSERT(LOOKUP(X,Y,RES)).

VERDES

:- DYNAMIC LOOKUP/3.

ADDANDSQUARE(X,Y,RES):-

LOOKUP(X,Y,RES), !.

ADDANDSQUARE(X,Y,RES):-

\+ LOOKUP(X,Y,RES), !,

RES IS (X+Y) * (X+Y),

ASSERT(LOOKUP(X,Y,RES)).

UNA PALABRA DE CUIDADO

- ALGUNAS REFLEXIONES SOBRE LA MANIPULACIÓN DE BASES DE DATOS:
 - A MENUDO UNA TÉCNICA ÚTIL
 - PERO PUEDE CONDUCIR A CÓDIGO SUCIO Y DIFÍCIL DE ENTENDER
 - ES NO DECLARATIVO, NO LÓGICO
 - POR LO TANTO, DEBE USARSE CON PRECAUCIÓN

DIFERENCIAS EN LA IMPLEMENTACIÓN

- LOS INTÉRPRETES DE PROLOG TAMBIÉN DIFIEREN EN LA FORMA EN QUE ASSERT Y RETRACT SE IMPLEMENTAN CON RESPETO AL BACTRAKING.
- LA OPERACIÓN DE ASSERT O RETRACT SON CANCELADO POR BACTRACKING.

RECOLECTEMOS SOLUCIONES

- AHORA PRESENTAMOS ALGUNOS PREDICADOS ÚTILES PARA PROCESAR TODAS LAS SOLUCIONES A UNA CONSULTA DE UNA SOLA VEZ.
- CHILD(MARTHA,CHARLOTTE).
- CHILD(CHARLOTTE,CAROLINE).
- CHILD(CAROLINE,LAURA).
- CHILD(LAURA,ROSE).
- DESCEND(X,Y):- CHILD(X,Y).
- DESCEND(X,Y):- CHILD(X,Z),
- DESCEND(Z,Y).
- ?- DESCEND(MARTHA,X).

DIFERENCIAS EN LA IMPLEMENTACIÓN

- PUEDE HABER MUCHAS SOLUCIONES A UNA CONSULTA DE PROLOG.
- SIN EMBARGO, PROLOG GENERA SOLUCIONES UNA POR UNA.
- A VECES NOS GUSTARÍA TENER TODO LAS SOLUCIONES A UNA CONSULTA DE UNA SOLA VEZ.
- NO HACE FALTA DECIR QUE SERÍA ÚTIL TENERLOS EN UN FORMATO ORDENADO Y UTILIZABLE.
- PROLOG TIENE TRES PREDICADOS INTEGRADOS : FINDALL/3, BAGOF/3 Y SETOF/3.
- EN ESENCIA, TODOS ESTOS PREDICADOS RECOPILAN TODAS LAS SOLUCIONES A UNA CONSULTA Y LAS PONEN EN UNA SOLA LISTA.
- PERO HAY DIFERENCIAS IMPORTANTES ENTRE ELLOS.

FINDALL(O,G,L)

- LA CONSULTA PRODUCE UNA LISTA L DE TODOS LOS OBJETOS O QUE SATISFACEN EL OBJETIVO G
- SIEMPRE TIENE ÉXITO UNIFICA L CON LISTA VACÍA SI G NO SE PUEDE SATISFACER.
- CHILD(MARTHA,CHARLOTTE).
- CHILD(CHARLOTTE,CAROLINE).
- CHILD(CAROLINE,LAURA).
- CHILD(LAURA,ROSE).
- DESCEND(X,Y):- CHILD(X,Y).
- DESCEND(X,Y):- CHILD(X,Z),
- DESCEND(Z,Y).

FINDALL(O,G,L)

- ?- FINDALL(X,DESCEND(MARTHA,X),L).
- ?- FINDALL(X,DESCEND(ROSE,X),L).
- ?- FINDALL(D,DESCEND(MARTHA,X),L).
- ?- FINDALL(CHI,DESCEND(MOT,CHI),L).

BAGOF(O,G,L)

- LA CONSULTA PRODUCE UNA LISTA **L** DE TODOS LOS OBJETOS **O** QUE SATISFAGAN EL OBJETIVO **G**.
- SÓLO TIENE ÉXITO SI EL OBJETIVO **G** TIENE ÉXITO.
- ENLAZA VARIABLES LIBRES EN **G**.
- ?- BAGOF(CHI,DESCEND(MOT,CHI),L).
- ?- BAGOF(CHI,MOT^DESCEND(MOT,CHI),L).

SETOF(O,G,L)

- LA CONSULTA PRODUCE UNA LISTA ORDENADA **L** DE TODOS LOS OBJETOS **O** QUE SATISFACEN EL OBJETIVO **G**.
 - SOLO TIENE ÉXITO SI EL OBJETIVO **G** TIENE ÉXITO.
 - ENLAZA VARIABLES LIBRES EN **G**.
 - ELIMINAR DUPLICADOS DE **L**.
 - ORDENA LAS RESPUESTAS EN **L**.
- ¿- SETOF(CHI,MOT^DESCEND(MOT,CHI),L).

EJERCITEMOS

- SUPONGAMOS QUE COMENZAMOS CON UNA BASE DE DATOS VACÍA. LUEGO DAMOS LA ORDEN:
 - `ASSERT(Q(A,B)), ASSERTZ(Q(1,2)), ASSERTA(Q(FOO,BLUG))`).
- ¿QUÉ CONTIENE AHORA LA BASE DE DATOS?
- LUEGO DAMOS LA ORDEN:
 - `RETRACT(Q(1,2)), ASSERTZ((P(X) :- H(X)))`.
- ¿QUÉ CONTIENE AHORA LA BASE DE DATOS?
- LUEGO DAMOS LA ORDEN:
 - `RETRACTALL(Q(_,_))`.
- ¿QUÉ CONTIENE AHORA LA BASE DE DATOS?

EJERCITEMOS

- SUPONGAMOS QUE TENEMOS LA SIGUIENTE BASE DE DATOS:
 - $Q(\text{BLOB}, \text{BLUG})$.
 - $Q(\text{BLOB}, \text{BLAG})$.
 - $Q(\text{BLOB}, \text{BLIG})$.
 - $Q(\text{BLAF}, \text{BLAG})$.
 - $Q(\text{DANG}, \text{DONG})$.
 - $Q(\text{DANG}, \text{BLUG})$.
 - $Q(\text{FLAB}, \text{BLOB})$.
- ¿CUÁL ES LA RESPUESTA DE PROLOG A LAS CONSULTAS?
 - $\text{FINDALL}(X, Q(\text{BLOB}, X), \text{LIST})$.
 - $\text{FINDALL}(X, Q(X, \text{BLUG}), \text{LIST})$.
 - $\text{FINDALL}(X, Q(X, Y), \text{LIST})$.
 - $\text{BAGOF}(X, Q(X, Y), \text{LIST})$.
 - $\text{SETOF}(X, Y^{\wedge}Q(X, Y), \text{LIST})$.

EJERCITEMOS

- ESCRIBE UN PREDICADO SIGMA/2 QUE TOMA UN ENTERO $N > 0$ Y CALCULE LA SUMA DE TODOS LOS ENTEROS DE 1 A N. POR EJEMPLO:
- ?- SIGMA(3,X).
X = 6
YES
- ?- SIGMA(5,X).
X = 15
YES
- ESCRIBA EL PREDICADO DE MODO QUE LOS RESULTADOS SE ALMACENEN EN LA BASE DE DATOS (NUNCA DEBE HABER MÁS DE UNA ENTRADA EN LA BASE DE DATOS PARA CADA VALOR) Y SE REUTILICEN SIEMPRE QUE SEA POSIBLE. POR EJEMPLO, SUPONGAMOS QUE HACEMOS LA SIGUIENTE CONSULTA:
- ?- SIGMA(2,X).
X = 3
YES
- ?- LISTING.
SIGMARES(2,3).
- ENTONCES, SI CONTINUAMOS PREGUNTANDO
- ?- SIGMA(3,X).
- PROLOG NO DEBE CALCULAR TODO LO NUEVO, SINO QUE DEBE OBTENER EL RESULTADO PARA SIGMA(2,3) DE LA BASE DE DATOS Y SOLO AGREGAR 3 A ESO. A CONTINUACIÓN, DEBERÍA RESPONDER: YES
?- LISTING.
SIGMARES(2,3).
SIGMARES(3,6).