

Programación recursiva en Prolog: Quadrees

Curso 2003/04, Fecha:19/12/2003

Enunciado 1 (quadrees) *Un quadtree puede representarse de forma recursiva como:*

- *Un cuadrado vacío mediante la constante vacío*
- *Un rectángulo de un color fijo, por ejemplo: `rect(rgb(1,0,0))`*
- *Una división en cuatro cuadrantes, por ejemplo: `d(vacio, rect(rojo), rect(verde), vacio)`*

El siguiente predicado `wqt(QT)` almacena el quadtree `QT` es un fichero VRML

```
wqt(QT):-wqt('qt.wrl',QT).
```

```
wqt(F,OT):-open(F,write,S),
             cabecera(S),
             viewPoint(S),
             wqt(S,0,0,64,OT),
             close(S).
```

```
wqt(S,-,-,-,vacio):-write(S,'').
```

```
wqt(S,X,Y,D,rect(C)):-
```

```
    putBox(S,X,Y,0,C,D,D,1).
```

```
wqt(S,X,Y,D,d(C1,C2,C3,C4)):-
```

```
    (D=<1->write(S,''))
```

```
    ; D2 is D/2, D4 is D/4,
```

```
      XP is X+D4, XN is X-D4,
```

```
      YP is Y+D4, YN is Y-D4,
```

```
      wqt(S,XN,YP,D2,C1),
```

```
      wqt(S,XP,YP,D2,C2),
```

```
      wqt(S,XN,YN,D2,C3),
```

```
      wqt(S,XP,YN,D2,C4)
```

```
    ).
```

```
cabecera(S):-
```

```
    write(S,'#VRML V2.0 utf8\n\n').
```

```
viewPoint(S):-
```

```

write(S, 'Viewpoint { description "Punto" \n'},
write(S, '          orientation 0 0 1 0\n'),
write(S, '          position 0 0.2 150 } \n').

putSphere(S,X,Y,Z,C,R):-
  translate(S,X,Y,Z),
  color(S,C), sphere(S,R), endColor(S),
  endTranslate(S).

putBox(S,X,Y,Z,C,SX,SY,SZ):-
  translate(S,X,Y,Z),
  color(S,C),
  box(S,SX,SY,SZ),
  endColor(S),
  endTranslate(S).

translate(S,X,Y,Z):-
  write(S, ' Transform { translation '), write3(S,X,Y,Z), nl(S),
  write(S, ' children [').

endTranslate(S):-
  write(S, ' ] }\n').

color(S,rgb(R,G,B)):-
  write(S, ' Shape { appearance Appearance '),
  write(S, ' { material Material { diffuseColor '),
  write3(S,R,G,B),
  write(S, ' } }').

endColor(S):-
  write(S, ' } ').

box(S,X,Y,Z):-
  write(S, ' geometry Box { size '),
  write3(S,X,Y,Z),
  write(S, ' } ').

sphere(S,R):-
  write(S, ' geometry Sphere { radius '),
  write(S,R),
  write(S, ' } ').

write3(S,X,Y,Z):-
  format(S, '~2f ~2f ~2f ',[X,Y,Z]).

  Un ejemplo de llamada sería ?-wqt(d(vacio,rect(rgb(1,0,0)), rect(rgb(0,1,0)), vacio)).

```

Construir diversos quadtrees

Nota: Si se utiliza el SWI-Prolog editor es necesario deshabilitar la opción Integrated Swi-Prolog window en el menú Configuration para que el programa anterior pueda realmente escribir en el fichero

Enunciado 2 (escal) Construir un predicado *escal(X)* que se cumple si *X* es un cuadtrees con forma de escalón. Un ejemplo puede verse en:

<http://www.di.uniovi.es/labra/PLF/prac/worlds/escal.wrl>.

Enunciado 3 (escalN) Construir un predicado *escalN(N,X)* que se cumple si *X* es un cuadtrees con forma de escalón con una resolución *N*. Un ejemplo con resolución 3 puede verse en:

<http://www.di.uniovi.es/labra/PLF/prac/worlds/escal3.wrl>.

Enunciado 4 (gira) Construir un predicado *gira(X,Y)* que se cumple si *Y* es el cuadtrees formado al girar *X* 90 grados.

Enunciado 5 (giros) Construir un predicado *giros(X,Y)* que se cumple si *Y* es un cuadtrees formado por repetir el cuadtrees *X* girando 90° cada cuadrante.

Enunciado 6 (romboN) Construir un predicado *romboN(N,X,Y)* que se cumple si *Y* es un cuadtrees en forma de rombo con una resolución *N*. Un ejemplo con resolución 3 puede verse en:

<http://www.di.uniovi.es/labra/PLF/prac/worlds/rombo3.wrl>.

Enunciado 7 (Galería) Construir una galería de quadtrees

Enunciado 8 (Colorea(Opcional)) Construir un predicado *colorea(A,B)*, que se cumple si *B* es un cuadtrees con el mismo formato que *A* pero coloreado de forma que ningún cuadrante adyacente tenga el mismo color. Por ejemplo, una posible solución al colorear un rombo puede verse en:

<http://www.di.uniovi.es/labra/PLF/prac/worlds/colorQT.wrl>.

Enunciado 9 (octrees(Opcional)) Tomando como referencia el predicado *wqt*, construir un predicado *wot* que permita representar octrees. Construir una galería con octrees generados en Prolog