

Eliminación de la recursividad mediante Pilas

Cada vez que se genera una llamada a una rutina recursiva se crea un *registro de activación* donde se almacenan las constantes, variables y parámetros por valor sobre las que se ejecutará esa copia del programa. Además, las llamadas recursivas se comportan como una pila: la última copia generada es la primera que se termina de ejecutar y además se ejecuta sobre los últimos valores generados para los parámetros. De forma que se comportan como si hubiésemos almacenado variables y parámetros en una pila, y luego fuésemos recuperando estos datos y ejecutando el procedimiento en cuestión sobre cada una de ellas (en realidad, esto es lo que ocurre). Así, para eliminar la recursividad y obtener una versión iterativa de un programa se puede seguir el proceso siguiente:

1. Mientras que los parámetros no se correspondan con el caso base, de forma iterativa:
 - a) Obtener los parámetros y variables locales sobre los que se ejecutaría cada copia recursiva y,
 - b) Almacenarlos en pilas separadas.
2. Mientras las pilas no estén vacías:
 - a) Sacar los parámetros y variables locales introducidos y,
 - b) Ejecutar sobre ellos el código que está después de la llamada recursiva.

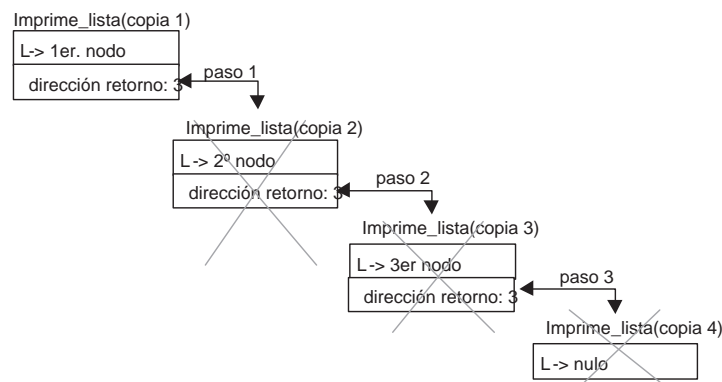
Ejemplo. Proceso recursivo que imprime en orden inverso una lista implementada de forma dinámica.

```

procedure Imprimir_lista_inversa (L: tLista);
{ Objetivo: imprimir el contenido de una lista a la inversa
  Entrada: L: inicio de la lista
  Salida: por pantalla }
begin
  1. if L<>nulo
  2. then begin
  3. Imprimir_lista_inversa (L^.sig);
  4. write(L^.info);
  5. end;
end;

```

Registros de activación y ejecución para una lista de tres nodos:



En este ejemplo, tenemos un único parámetro —un puntero a un nodo de la lista— y ninguna variable local. Creamos la versión iterativa mediante el uso de pilas.

```

procedure Imprimir_lista_inversa (L: tLista);

uses    TDA_Pila; {con tInfo=tLista}

var    Pila: tPila; begin
        PilaVacia(Pila);

        {Vamos almacenado los valores de los parámetros en la pila}
        while L<>nulo do
        begin
            Meter(Pila, L);
            L:= L^.sig;
        end;

        {Ya están todos los parámetros introducidos en la pila}
        {ejecutamos las instrucciones después de l llamada recursiva}
        while not EsPilaVacia(Pila) do
        begin
            Cima (Pila,L);
            quitar(Pila);
            write(L^.info);
        end;
end;

```