

ENUNCIADO

Los cuerpos de las serpientes estan formados por anillos. En la víbora de Montegancedo estos anillos siguen un patrón de color que se repite a lo largo del cuerpo. Si se representan los colores por letras, el cuerpo de una víbora de Montegancedo con 15 anillos que siguen el patrón abcd tiene el aspecto de la secuencia abcdabcdabcdabc.

Para dormir, la vibora de Montegancedo recoge el cuerpo en forma de rectángulo: estira el cuerpo hacia la derecha hasta un punto en que se dobla y lo estira de nuevo hacia la izquierda, paralelo a sí misma, hasta llegar a la cabeza, se vuelve a doblar sobre sí misma y así sucesivamente. La víbora anterior tendría el siguiente aspecto mientras duerme:

```
abcd  
badcb  
cdabc
```

La cabeza es el primer anillo 'a', el cuerpo continua hacia la derecha hasta la siguiente 'a', se dobla en la 'b' que hay debajo y continua hacia la izquierda. Y así sucesivamente.

Se pide al alumno programar un predicado `vibora/4` cuyo cuarto argumento sea el rectangulo que forma una víbora de Montegancedo que sigue el patron del primer argumento (en forma de lista) la cual se recoge para dormir formando filas con tantos anillos como elementos tiene la lista del segundo argumento y columnas con tantos anillos como elementos tiene la lista del tercer argumento. El rectángulo se representa con una lista de listas, cada una de las cuales representa una fila del rectángulo con los colores de los anillos correspondientes.

Por ejemplo, para el caso de la vibora mencionada mas arriba:

?- `vibora([a,b,c,d],[_,_,_,_],[_,_,_],R)`.

`R = [[a,b,c,d,a],[b,a,d,c,b],[c,d,a,b,c]]` ?

Nótese que el patrón de una fila (segundo argumento) indica cuántas columnas tiene el rectángulo y el patrón de una columna (tercer argumento) indica el número de filas que tiene el rectángulo. Puesto que no se especifica la longitud de la víbora, se supone que es igual al producto de las longitudes del segundo y tercer argumento, por lo que siempre "rellena" todo el rectángulo.

El programa debe de ser puro, pero no se puede utilizar aritmética de Peano. Debe funcionar en cualquier modo de llamada siempre que segundo y tercer argumentos sean listas completas no vacías, es decir, con un número exacto de elementos distinto de cero. Los elementos de estas listas son indiferentes. Por ejemplo, las llamadas "admisibles" pueden tener en dichos argumentos listas como `[_,_,_]` o `[a,a,a,a,a]` o `[0,X,a,p(b)]` pero no `[_X]` ni `X` si `X` es una variable libre. El programa puede no funcionar correctamente con llamadas "no admisibles" como las mencionadas, pero debe hacerlo en cualquier otro caso.