

ACTIVIDAD 3

IMPLEMENTACIÓN de Heurísticas y Backtracking

1. Sea la siguiente formulación del problema del coloreado de las regiones de Australia en Matlab. Completa la matriz de restricciones de acuerdo al mapa del problema:

```
function psr=crea_PintaRegiones_psr
%%% psr es una estructura con los siguientes elementos
%%% asignacion: vector columna cada fila corresponde con el valor asignado
%%% a la variable(i). Inicialmente debe estar a un valor imposible
%%% dominios: matriz donde cada fila representa los valores discretos de
%%% cada variable
%%% restricciones: matriz donde se indica las regiones adyacentes con 1

N=7;
Variables=1:N; %% 1 2 3 4 5 6 7
asignacion=zeros(1,N); %% AO TN AS Q NGS V T
dominios=repmat([1 2 3],N,1); % 1 Rojo 2 Verde 3 Azul

%%% COMPLETAR la matriz de restricciones

restricciones=zeros(N,N);
restricciones(1,[2:3])=1; %% AO con TN y AS
restricciones(2, )=1;
restricciones(3, )=1;
restricciones(4, )=1;
restricciones(5, )=1;
restricciones(6, )=1;

psr=struct('variables',variables, ...
          'asignacion',asignacion,...
          'dominios', dominios, ...
          'restricciones',restricciones);
```

2. Función **isSafe**: esta función puede ser implementada en Matlab de una forma más óptima. Utiliza la estructura psr y optimiza esta función:

```
function s = isSafe(k,c,n,G,x)

s=1;
i=1;
while(i<=n && s==1)
    if(G(k,i)== 1 && c == x(i))
        s=0;
    end
    i=i+1;
end
end
```

3. Completa las siguientes funciones en Matlab para obtener las heurísticas estudiadas en este tema. El primer caso en cada función correspondería a no aplicar ninguna heurística, simplemente tomar la siguiente variable o el siguiente valor no asignado todavía.

```
function [vari]=selecciona_var_no_asignada(psr, caso)
vari=0;
noasig=find(psr.asignacion==0);
asig=find(psr.asignacion>0);
if length(noasig)>0
    switch(caso)
        case 1,
            %%% Asigna la siguiente variable no asignada

        case 2,
            %%% Grado Heurístico: Asigna variable que participa en más
            restricciones

        case 3,
            %%% MVR: Asigna la variable con menos valores restantes

        otherwise,
            vari=0;
    end
end

function valores=orden_valores_dominio(vari, psr, caso)
%%%function orden_valores_dominio(vari, psr)

valores=0;
posibles=.. %%% indices posibles

if length(posibles)>0
    switch(caso)
        case 1,
            %%% Asigna el siguiente valor no asignado

        case 2,
            %%% Valor presente en el dominio del menor numero de variables

        otherwise,
            valores=0;
    end
end
```

4. Incorpora estas funciones al algoritmo de backtracking que ya has implementado y comprueba que todo funciona correctamente.