

Práctica 10 – Programación con Restricciones – Jacop

Preparación:

- 1.- Bajarse el archivo 'practica10.zip' y descomprimirlo
- 2.- También bajarse y copiar en c:\hlocal ***jacop-4.6.jar***
- 3.- Desde Eclipse importar la carpeta practica10 como proyecto existente

Enunciado:

Vamos a representar un grafo mediante sus aristas. Cada arista viene representada por el su vértice origen y su vértice fin.

1. [0.5] Declarar el almacén (Store) como variable estática.
2. [0.5] Declarar dos arrays static de variables enteras Jacop (IntVar) ini y fin, ambos de tamaño numAristas. Contendrán el principio y el fin de cada arista
3. Inicializar ambos arrays. Cada variable (IntVar) contendrá un valor entre 1 y numAristas y tendrá como nombre "ini0, ini1, ..." (para el caso de las variables en el array ini) o "fin0, fin1, ...". La solución será como:
`ini0=1, ini1=1, ini2=1, ini3=1, ini4=1, fin0=1, fin1=1, fin2=1, fin3=1, fin4=1]`
4. Ahora añadir restricciones para asegurar que no hay bucles, es decir que para todo i
`ini[i] != fin[i]`
5. Ahora usar alguna constraint global para asegurar que ini contiene valores distintos. Hacer lo mismo con fin. Una posible salida:
`[ini0=1, ini1=2, ini2=3, ini3=4, ini4=5, fin0=2, fin1=5, fin2=1, fin3=3, fin4=4]`
6. Añadir restricciones para que en fin los valores queden ordenados de forma decreciente. Una posible salida:
`[ini0=1, ini1=2, ini2=4, ini3=3, ini4=5, fin0=5, fin1=4, fin2=3, fin3=2, fin4=1]`

Primitive Constraint	JaCoP specification
$X = Const$	XeqC(X, Const)
$X = Y$	XeqY(X, Y)
$X \neq Const$	XneqC(X, Const)
$X \neq Y$	XneqY(X, Y)
$X > Const$	XgtC(X, Const)
$X > Y$	XgtY(X, Y)
$X \geq Const$	XgteqC(X, Const)
$X \geq Y$	XgteqY(X, Y)
$X < Const$	XltC(X, Const)
$X < Y$	XltY(X, Y)
$X \leq Const$	XlteqC(X, Const)
$X \leq Y$	XlteqY(X, Y)
$X \cdot Const = Z$	XmulCeqZ(X, Const, Z)
$X + Const = Z$	XplusCeqZ(X, Const, Z)
$X + Y = Z$	XplusYeqZ(X, Y, Z)
$X + Y + Const = Z$	XplusYplusCeqZ(X, Y, Const, Z)
$X + Y + Q = Z$	XplusYplusQeqZ(X, Y, Q, Z)
$X + Const \leq Z$	XplusClteqZ(X, Const, Z)
$X + Y \leq Z$	XplusYlteqZ(X, Y, Z)
$X + Y > Const$	XplusYgtC(X, Y, Const)
$X + Y + Q > Const$	XplusYplusQgtC(X, Y, Q, Const)
$ X = Y$	AbsXeqY(X, Y)
Non-primitive Constraint	JaCoP specification
$X \cdot Y = Z$	XmulYeqZ(X, Y, Z)
$X \div Y = Z$	XdivYeqZ(X, Y, Z)
$X \bmod Y = Z$	XmodYeqZ(X, Y, Z)
$X^Y = Z$	XexpYeqZ(X, Y, Z)