

# Programación Lógica

## Laboratorio 2 – 2018

**Facultad de Ingeniería  
Instituto de Computación  
Grupo de Procesamiento de Lenguaje Natural**

*El objetivo de este obligatorio es implementar un puzzle aritmético llamado Kakuro en Prolog y en Prolog + CLP(FD) (Programación Lógica con Restricciones).*

### **Nota previa - IMPORTANTE**

Se debe cumplir íntegramente el "Reglamento del Instituto de Computación ante Instancias de No Individualidad en los Laboratorios", disponible en:

<http://www.fing.edu.uy/inco/pm/uploads/Ense%flanza/NoIndividualidad.pdf>

En particular está prohibido utilizar documentación de otros grupos o de otros años, de cualquier índole, o hacer público código a través de cualquier medio (EVA, correo, papeles sobre la mesa, etc.).

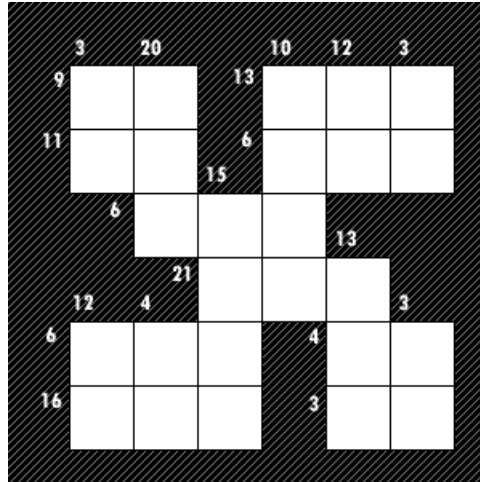
### **Kakuro**

El Kakuro es un puzzle aritmético similar a las palabras cruzadas, pero con sumas de números en lugar de definiciones. El nombre kakuro viene del japonés (es la abreviación japonesa de kasan kurosu: 加算クロス Adición+Cruz) y es un sucesor del *sudoku*, con bastante éxito actualmente.

El puzzle consiste en rellenar las casillas vacías de un tablero con los números de 1 al 9.

El tablero es rectangular, de dimensiones variables, y tiene 3 tipos de casillas: casillas blancas o vacías, que se deben rellenar durante el juego; casillas negras, que no tienen información y casillas con indicaciones de sumas. Los dos últimos tipos de casilla no se modifican durante el juego. Un grupo de casillas blancas consecutivas forma un bloque, ya sea en una fila o en una columna. Cada uno de estos bloques debe estar precedido por una casilla con suma. La suma indica la suma de la fila, si se encuentra a la izquierda de esta, o la suma de la columna, si se encuentra arriba de ella. Los números en una misma suma (en un mismo bloque) no deben repetirse. Por ejemplo si la suma de dos casillas es 16 en una casilla irá el 9 y en la otra irá el 7 no siendo admisible que ambas casillas se rellenen con un 8.

En la figura 1 se muestra un ejemplo de tablero inicial para un Kakuro cuadrado de 7x7.



*Ilustración 1: Tablero inicial de Kakuro de 7x7*

### Requerimientos a implementar

Se debe implementar en Prolog y en Prolog+CLP(FD) el puzzle Kakuro en modalidad jugador humano y en modalidad resolución por máquina. También se debe generar de modo aleatorio tableros de kakuro que luego se ofrecerán para resolución.

Más concretamente, se pide implementar:

**1-** un predicado Prolog `kalog(+NroJuego)` que despliegue el tablero de número `NroJuego` y maneje la interacción del juego con una interfaz gráfica en XPCE. En la interfaz se ofrecerá además la posibilidad de verificar la correctitud de las decisiones del jugador. Se suministrarán los predicados básicos de la interfaz gráfica. Se parte de la base de que en la Base de Datos Prolog están definidos los tableros con los cuales se va a realizar las pruebas. Se utiliza para ello un predicado

`juego(+NroJuego,+Tablero).`

Tablero es una matriz representada como una lista de filas, que son a su vez listas. Cada elemento de Tablero puede ser de alguna de las siguientes formas:

- casilla negra – `n`
- casilla cabeza de bloque de fila – `f(Suma)`
- casilla cabeza de bloque de columna – `c(Suma)`
- casilla que encabeza bloques de Fila y de Columna – `p(SumaFila,SumaColumna)`
- casilla vacía ( es la que se debe rellenar) – una variable Prolog

**2-** un predicado Prolog `kalog(+NroJuego,+Tecnica)` que represente la resolución del juego por la máquina, en 2 modalidades:

Tecnica = `std` - se utiliza Prolog estándar

Tecnica = clpfd – se utiliza Prolog + CLP(FD)

**3-** un predicado Prolog `kalog(+Filas,+Columnas,-Tablero)` que genera aleatoriamente tableros de Kakuro de las dimensiones especificadas. Se debe generar tableros que tengan solución. Los tableros que se generen deben tener la estructura definida en el punto 1.

## Insumos

Se proveerán los siguientes módulos Prolog:

- **graficos.pl**: Contiene todos los predicados para el manejo gráfico y de interacción con usuario. Se recomienda leer la especificación de los predicados exportados por este módulo para entender el funcionamiento de las funcionalidades gráficas. Este módulo no debe ser modificado.

- **kalog.pl**: Módulo principal de la solución. Se provee una implementación básica que muestra los principios de interacción con **graficos.pl**. Este módulo debe modificarse para implementar la solución del obligatorio.

## Observaciones

La implementación debe realizarse de manera que pueda ser ejecutada en la plataforma SWI-Prolog. Se pide específicamente la utilización de CLP en al menos una parte del laboratorio, pero se puede utilizar en general cuando se estime conveniente (excepto obviamente en el punto 2, en la parte en la que se pide una resolución en Prolog estándar).

## Forma de entrega

La entrega se realizará a través del espacio eva del curso. Se debe entregar un solo archivo '**grupo##.zip**', donde **##** es el número del grupo que realiza la entrega, conteniendo todos los módulos de la solución y el informe de la misma, ítems indicados en el apartado **Entregable**.

## Fecha de entrega

Los trabajos deberán ser entregados siguiendo el procedimiento descrito anteriormente antes del domingo 17/6/2016 a las 23:55, sin excepciones. No se aceptará ningún trabajo pasada la citada fecha.

## Entregable

El archivo a entregar debe contener:

1. Implementación del puzzle Kakuro, con todos los módulos necesarios para la ejecución del juego y específicamente los predicados pedidos. Incluir también el módulo **graficos.pl** y los archivos de imágenes. Comentar cada uno de los predicados definidos.

2. Informe en formato pdf detallando la estructura de los módulos, los predicados principales y las decisiones de diseño tomadas. En particular, se debe incluir además una comparación de las versiones Prolog estándar y CLP(FD) del punto 2, tanto en lo que refiere a la “declaratividad” de la

solución como al análisis de eficiencia.

## **Referencias**

[1] <http://www.sudokumania.com.ar/juegos/kakuro#reglas>

[2] <http://www.swi-prolog.org/man/clpfd.html>