

Programación lógica y funcional

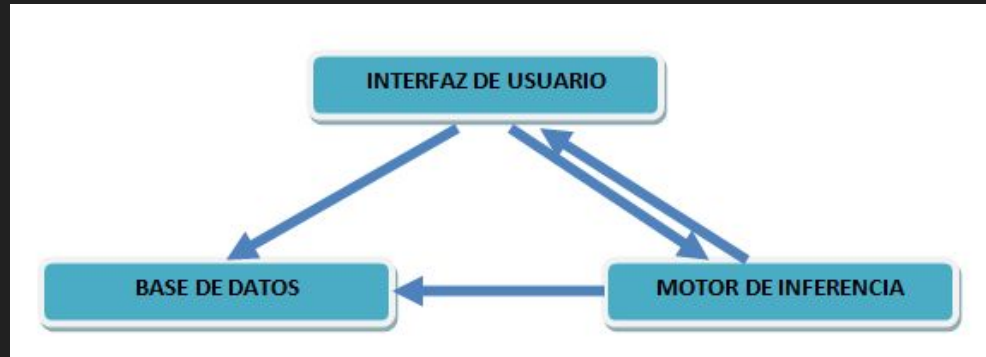
Integrantes:

Conti, Bruno Nicolás - bnconti@comunidad.unnoba.edu.ar

Polo, Marcos Alexis - mapolo@comunidad.unnoba.edu.ar

Sistema experto diagnosticador

El diseño de la solución está separado en tres partes claramente definidas



SISTEMA EXPERTO DIAGNOSTICADOR

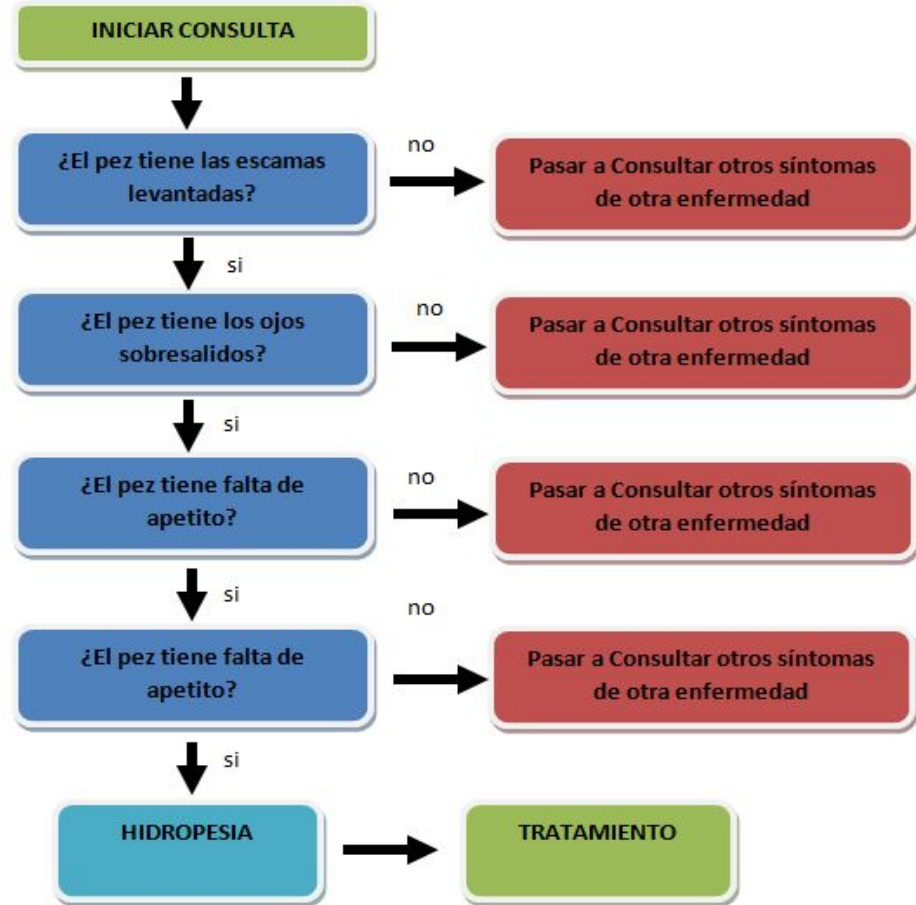
Sistema experto diagnosticador

En el desarrollo de este proyecto implementaremos un SE que ayude a las personas a diagnosticar las enfermedades de los peces goldfish.

Mediante el mismo podrán conocer la enfermedad que afecta al pez y obtener también un tratamiento.

Sistema experto diagnosticador

- Lógica del motor de inferencia para el caso de la hidropesía.
- Esta parte del sistema se encarga de decidir cuál será el diagnóstico a partir de las preguntas realizadas al usuario



Sistema experto diagnosticador

Si bien el código es algo extenso debido al manejo de las interfaces de usuario, el núcleo de la lógica es más acotado y puede resumirse de la siguiente manera.

1. El programa toma una enfermedad y pregunta si el pez sufre los síntomas que le corresponden a esta.
 - a. Si responde SI: el programa continúa consultando por los demás preguntas de la misma enfermedad.
 - b. Si responde NO: se pasa a la siguiente enfermedad y comienza a preguntar de nuevo por los síntomas de la nueva.
2. El programa finaliza cuando se responda afirmativamente a todos los síntomas de cierta enfermedad o nos hayamos quedado sin ninguna que comprobar en nuestra base de hechos.

Sistema experto diagnosticador

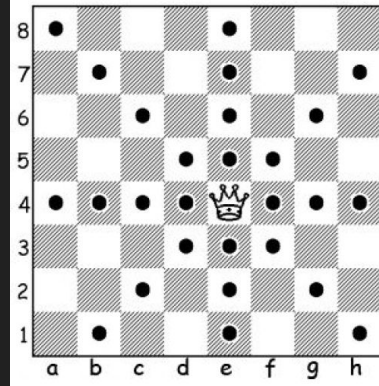
CONCLUSIONES Y MEJORAS A FUTURO

- Una mejora que aumentaría drásticamente la calidad de diagnóstico es que funcione de un modo probabilístico.
- ¿Pero qué ocurre cuando de los cuatro síntomas que presenta una enfermedad solamente tiene tres?
- ¿O si tiene esos tres, y otro síntoma correspondiente a otra enfermedad?
- Es en casos como estos, donde no se puede determinar una única respuesta con 100% de seguridad, donde encontramos las mayores debilidades del sistema.

PROBLEMA DE LAS N REINAS

Problema de las n reinas

- ¿De qué trata?
- Formas en las que se presenta
 - Distintos tamaños de n
 - Buscar una solución o todas
 - Con algunas piezas ya colocadas
- Con piezas ya colocadas es NP completo.
- La cant. de soluciones incrementa exponencialmente.
- Siempre hay solución para $n > 3$.

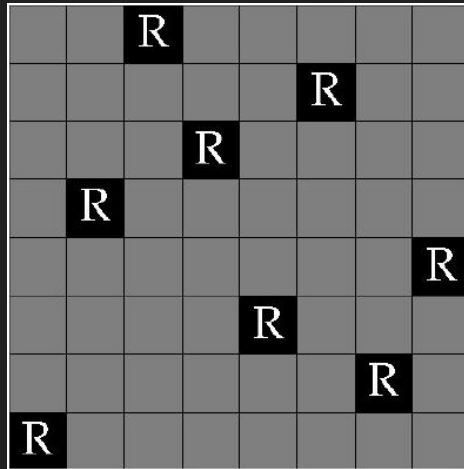


| n | $R(n)$ |
|-----|-----------|
| 5 | 10 |
| 6 | 4 |
| 7 | 40 |
| 8 | 92 |
| 9 | 352 |
| 10 | 724 |
| 11 | 2.680 |
| 12 | 14.200 |
| 13 | 73.712 |
| 14 | 365.596 |
| 15 | 2.279.184 |

Para $n = 27$, $R(n) =$
234.907.967.154.122.528

Problema de las n reinas

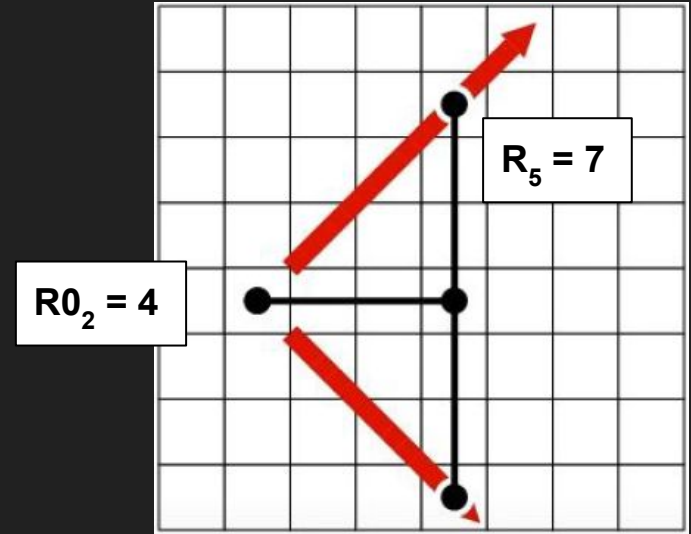
- ¿Cómo se representa el tablero en el programa?
- Ejemplo: [1,5,8,6,3,7,2,4] (solución para $n = 8$).



Problema de las n reinas

- Cláusula principal:

```
reinas_seguras_aux([R|Rs], R0, D0) :-  
    R #\= R0,  
    abs(R0 - R) #\= D0,  
    D1 #= D0 + 1,  
    reinas_seguras_aux(Rs, R0, D1).
```



Problema de las n reinas

- Ejemplos de ejecución

```
?- n_reinas(8, Rs), labeling([ff], Rs).  
   Rs = [1,5,8,6,3,7,2,4]  
;   Rs = [1,6,8,3,7,4,2,5]  
;   ...
```

```
?- n_reinas(N, [2,4,1,3]).  
   N = 4  
;   ...  
?- n_reinas(N, [2,4,C,D]).  
   N = 4, C = 1, D = 3  
;   ...
```

```
?- n_reinas(N, Rs), labeling([ff], Rs).  
   N = 0, Rs = []  
;   N = 1, Rs = [1]  
;   N = 4, Rs = [2,4,1,3]  
;   N = 4, Rs = [3,1,4,2]  
;   N = 5, Rs = [1,3,5,2,4]  
;   N = 5, Rs = [1,4,2,5,3]  
;   N = 5, Rs = [2,4,1,3,5]  
;   N = 5, Rs = [2,5,3,1,4]  
;   N = 5, Rs = [3,1,4,2,5]  
;   N = 5, Rs = [3,5,2,4,1]  
;   N = 5, Rs = [4,1,3,5,2]  
;   N = 5, Rs = [4,2,5,3,1]  
;   N = 5, Rs = [5,2,4,1,3]  
;   N = 5, Rs = [5,3,1,4,2]  
;   N = 6, Rs = [2,4,6,1,3,5]  
;   N = 6, Rs = [3,6,2,5,1,4]  
;   N = 6, Rs = [4,1,5,2,6,3]  
;   N = 6, Rs = [5,3,1,6,4,2]  
;   N = 7, Rs = [1,3,5,7,2,4,6]  
;   N = 7, Rs = [1,4,7,3,6,2,5]  
;   ...
```

INFORME DE HOROSCOPO

Informe de horoscopo

En este informe se presenta el desarrollo de un sistema que permite al usuario conocer su signo zodiacal basándose en su fecha de nacimiento.

Informe de horoscopo

PROCESO

Para entender mejor este proceso, mostraremos el siguiente ejemplo acompañado del árbol de decisión que se crea a partir de la consulta, la cual es:

`signo(14, 06, Signo)` .

Para que esto sea verdad, se deben encontrar los valores de Signo que cumplan las restricciones impuestas, es decir, Prolog tomará la regla signo y buscará en su base de conocimientos los signos que cumplan la condición de mes 6 y día 14.

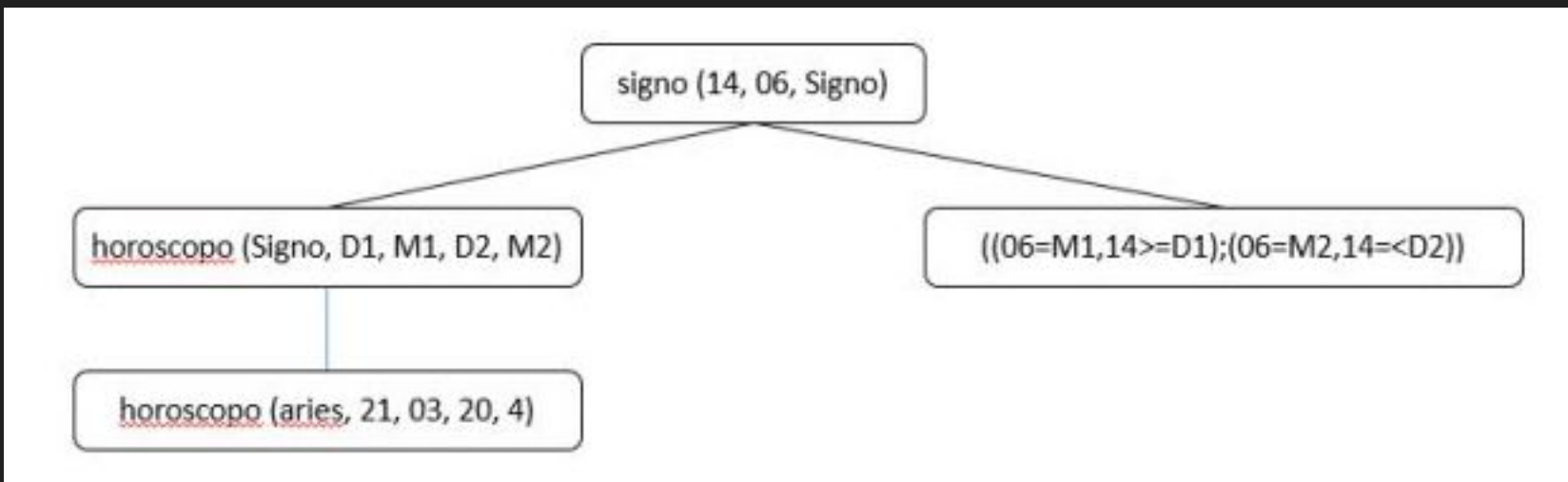
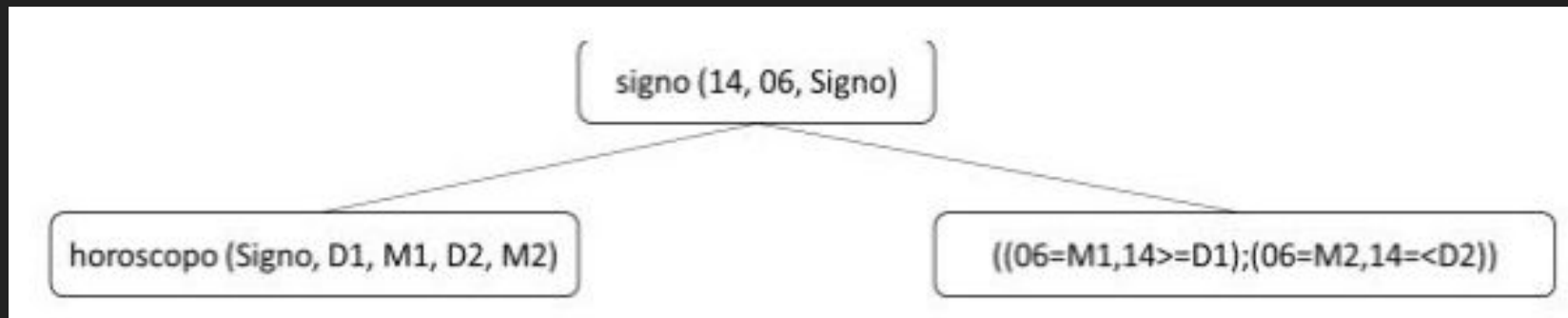
Informe de horoscopo

Definimos los hechos para cada signo zodiacal en el predicado horoscopo, y en cada uno de sus cláusulas definiremos un signo del zodíaco.

Como vemos, las cláusulas siguen el siguiente formato: horóscopo(nombre signo, día inicio, mes inicio, día fin, mes fin).

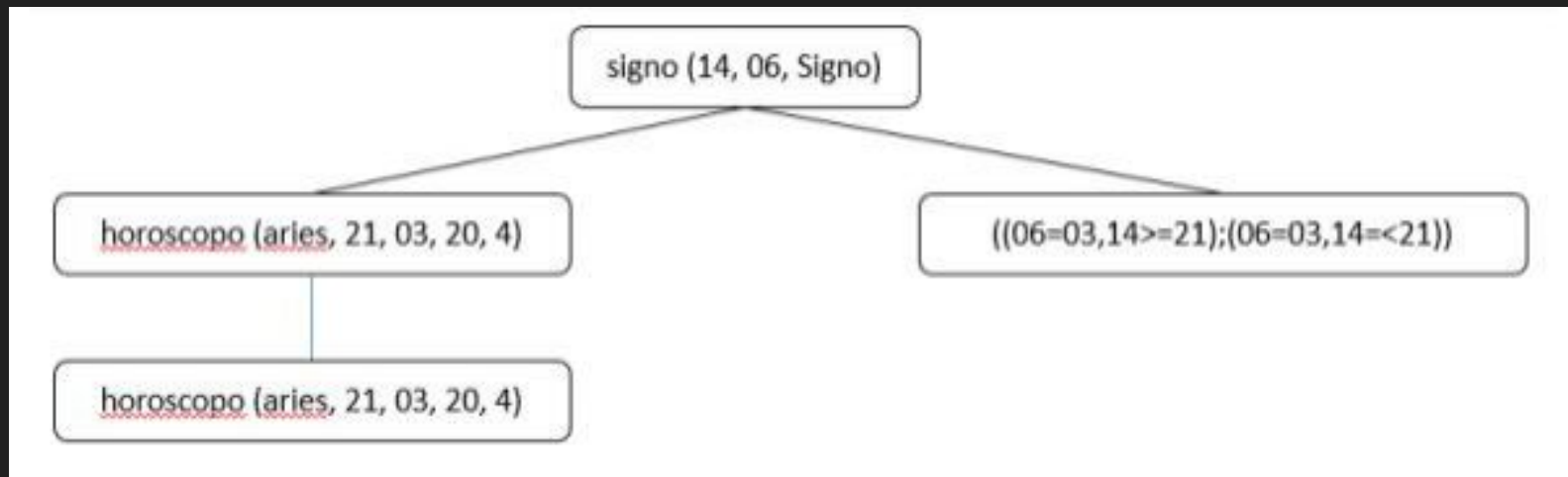
```
horoscopo(aries,21,3,20,4).  
horoscopo(tauro,21,4,21,5).  
horoscopo(geminis,22,5,21,6).  
horoscopo(cancer,22,6,22,7).  
horoscopo(leo,23,7,22,8).  
horoscopo(virgo,23,8,22,9).  
horoscopo(libra,23,9,22,10).  
horoscopo(escorpio,23,10,21,11).  
horoscopo(sagitario,22,11,21,12).  
horoscopo(capricornio,22,12,20,1).  
horoscopo(acuario,21,2,20,3).
```


Informe de horoscopo



Informe de horoscopo

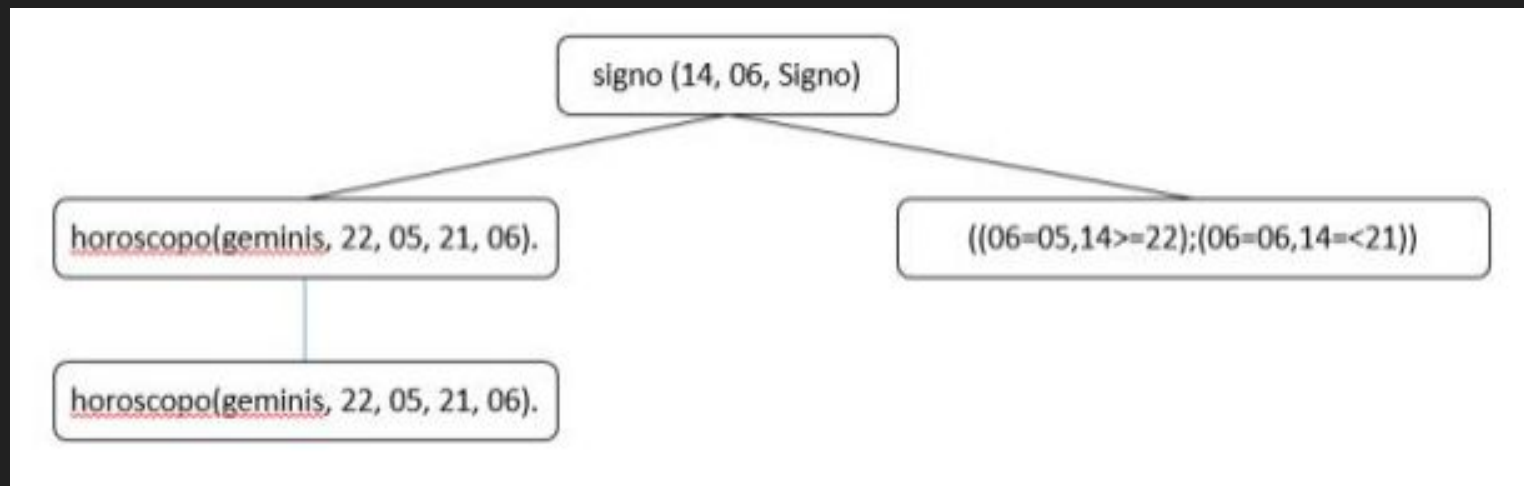
De esta forma obtenemos de la primera hipótesis una serie de valores que nos van a servir para cargar la segunda parte del predicado:



Entonces llegamos a obtener valores verdaderos para la primera rama del árbol, pero no para la segunda, ya que al compararlos con los valores ingresados por el usuario (mes 6, día 14) vemos que pueden compararse exitosamente.

Informe de horoscopo

Entonces esto que vimos anteriormente nos hará replantear nuevamente la consulta hasta que encuentre una solución exitosa o hasta quedarnos sin opciones de búsqueda



Eventualmente llegaremos al hecho horoscopo(geminis,22,05,21,06), esto es: Signo=geminis, D1=22, M1=5, D2=21, M2=6. Al cargar estos valores en la segunda rama obtenemos el siguiente árbol:

Informe de horoscopo

RESULTADOS

Y como pudimos ver, encontramos un éxito en la segunda rama, entonces, siendo la primera y segunda rama verdadera, se tiene un signo el cual satisface la hipótesis: Géminis.

Informe de horoscopo

Una vez terminados los hechos partimos de la hipótesis de que el signo de un día y mes sea verdadero si se cumple algún hecho definido de horóscopo, que el mes ingresado en la hipótesis sea igual al mes inicial del hecho, y el día ingresado mayor o igual al día inicial del hecho; o sino, que el mes ingresado sea igual al mes final del hecho y que el día ingresado sea menor o igual día final del hecho.

```
signo(Dia,Mes,Signo) :-  
    horoscopo(Signo,D1,M1,D2,M2),  
    ((Mes=M1, Dia>D1); (Mes=M2,Dia=<D2)).
```

Informe de horoscopo

CONCLUSIONES Y MEJORAS

Aquí se realiza backtracking hasta finalizar con los hechos para comprobar si no existe otro caso de éxito. Como vimos, en Prolog se utiliza la inferencia de encadenamiento hacia atrás, este parte desde una hipótesis creando ramas con las distintas reglas que se tengan hasta llegar a los hechos que se cumplan y comprobando que no existen más que si la cumplan.

Lo que nos pareció agregar a futuro sería una base de conocimientos donde además del signo tambien te devuelva una predicción a futuro del mismo o compatibilidades entre ellos.

SOLUCIONADOR DE SUDOKUS

Solucionador de sudokus

- ¿De qué trata?
- Términos: pista, unidad, etc.
- Solución implementada con Clojure.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 5 | 3 | | | 7 | | | | |
| 6 | | | 1 | 9 | 5 | | | |
| | 9 | 8 | | | | | 6 | |
| 8 | | | | 6 | | | | 3 |
| 4 | | | 8 | | 3 | | | 1 |
| 7 | | | | 2 | | | | 6 |
| | 6 | | | | | 2 | 8 | |
| | | | 4 | 1 | 9 | | | 5 |
| | | | | 8 | | | 7 | 9 |

| | | | | | | | | | |
|--------|----|----|----|----|----|----|----|----|-----------|
| A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | - Fila |
| B1 | B2 | B3 | B4 | B5 | B6 | B7 | B8 | B9 | - Columna |
| C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | - Región |
| -----+ | | | | | | | | | |
| D1 | D2 | D3 | D4 | D5 | D6 | D7 | D8 | D9 | |
| E1 | E2 | E3 | E4 | E5 | E6 | E7 | E8 | E9 | |
| F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 | |
| -----+ | | | | | | | | | |
| G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G9 | |
| H1 | H2 | H3 | H4 | H5 | H6 | H7 | H8 | H9 | |
| I1 | I2 | I3 | I4 | I5 | I6 | I7 | I8 | I9 | |

Solucionador de sudokus

Descripción general de la solución:

1. Inicializar la grilla del programa.
2. Si una casilla tiene un único valor posible, eliminarlo de sus pares.
3. Si a una unidad se le puede asignar un valor en una única casilla, colocarlo allí.
4. Volver a 2 y repetir hasta dar con la solución gracias al efecto dominó.

```
(def sudoku-dificil  
"  
850002400  
720000009  
004000000  
000107002  
305000900  
040000000  
000080070  
017000000  
000036040  
")
```

```
{  
  A1 = {8},  
  A2 = {5},  
  A3 = {1, 2, ..., 8, 9},  
  A4 = {1, 2, ..., 8, 9},  
  A5 = {1, 2, ..., 8, 9},  
  A6 = {2},  
  ...  
}
```

Solucionador de sudokus

Pero esos principios no alcanzan para los sudokus difíciles:

| | | | | | | | | |
|-------|-------|---------|-------|--------|--------|-------|-------|--------|
| 4 | 1679 | 12679 | 139 | 2369 | 269 | 8 | 1239 | 5 |
| 26789 | 3 | 1256789 | 14589 | 24569 | 245689 | 12679 | 1249 | 124679 |
| 2689 | 15689 | 125689 | 7 | 234569 | 245689 | 12369 | 12349 | 123469 |
| 3789 | 2 | 15789 | 3459 | 34579 | 4579 | 13579 | 6 | 13789 |
| 3679 | 15679 | 15679 | 359 | 8 | 25679 | 4 | 12359 | 12379 |
| 36789 | 4 | 56789 | 359 | 1 | 25679 | 23579 | 23589 | 23789 |
| 289 | 89 | 289 | 6 | 459 | 3 | 1259 | 7 | 12489 |
| 5 | 6789 | 3 | 2 | 479 | 1 | 69 | 489 | 4689 |
| 1 | 6789 | 4 | 589 | 579 | 5789 | 23569 | 23589 | 23689 |

5. Utilizar casillas pivote. Si se llega una solución válida: perfecto, y sino hacer backtracking y volver a probar con otro de los posibles valores.

Solucionador de sudokus

:: Problema:

```
850002400
720000009
004000000
000107002
305000900
040000000
000080070
017000000
000036040
```

:: Solución:

| | | | | | | | | | | |
|-------------------|---|---|--|---|---|---|--|---|---|---|
| 8 | 5 | 9 | | 6 | 1 | 2 | | 4 | 3 | 7 |
| 7 | 2 | 3 | | 8 | 5 | 4 | | 1 | 6 | 9 |
| 1 | 6 | 4 | | 3 | 7 | 9 | | 5 | 2 | 8 |
| -----+-----+----- | | | | | | | | | | |
| 9 | 8 | 6 | | 1 | 4 | 7 | | 3 | 5 | 2 |
| 3 | 7 | 5 | | 2 | 6 | 8 | | 9 | 1 | 4 |
| 2 | 4 | 1 | | 5 | 9 | 3 | | 7 | 8 | 6 |
| -----+-----+----- | | | | | | | | | | |
| 4 | 3 | 2 | | 9 | 8 | 1 | | 6 | 7 | 5 |
| 6 | 1 | 7 | | 4 | 2 | 5 | | 8 | 9 | 3 |
| 5 | 9 | 8 | | 7 | 3 | 6 | | 2 | 4 | 1 |

¡Muchas gracias!