

Descripcion del Trabajo

Nuestro modelo esta formado por **fichas, números y vacío**.

Un numero puede ser utilizado para representar una única coordenada o una distancia dentro del tablero. La distancia entre fichas se mide así: si dos fichas están juntas la distancia entre ellas es 1. Cualquiera sea el caso (que represente una distancia o coordenada), un número deberá ser siempre menor estricto que 8 y mayor o igual a 0 ($0 \leq N < 8$).

Funciones que se utilizan

- **getFicha(numero, numero)** *Dado un par de coordenadas retorna la ficha que se encuentra en esa posición o vacío si no hay ninguna ficha.*

Predicados que se utilizan

De identificación

- **esPeon(ficha)** *Es valido si ficha es un Peón.*
- **esAlfil(ficha)** *Es valido si ficha es un Alfil.*
- **esTorre(ficha)** *Es valido si ficha es una Torre.*
- **esCaballo(ficha)** *Es valido si ficha es un Caballo.*
- **esReina(ficha)** *Es valido si ficha es una Reina.*
- **esRey(ficha)** *Es valido si ficha es un Rey.*
- **esBlanca(ficha)** *Es valido si ficha es Blanca.*
- **esNegra(ficha)** *Es valido si ficha es Negra.*
- **estaVacio(numero, numero)** *Es valido si en la posición no hay fichas.*

De posición

- **mismaFila(ficha, ficha)** *Es valido si las fichas están en la misma fila, sin importar distancia o si se interponen otras fichas.*
- **mismaColumna(ficha, ficha)** *Es valido si las fichas están en la misma columna, sin importar distancia o si se interponen otras fichas.*
- **mismaDiagonal(ficha, ficha)** *Es valido si las fichas están en la misma diagonal, sin importar distancia o si se interponen otras fichas.*
- **estaEnL(ficha, ficha)** *Es valido si las fichas están en L. Esto quiere decir que una se encuentre de la otra 2 casilleros hacia arriba o hacia abajo y un casillero hacia los costados; hacia la izquierda o derecha y un casillero hacia arriba o abajo. Sin importar si se interponen otras fichas.*

De distancia

- **distancia(ficha, ficha, numero)** Es valido si la distancia entre las 2 fichas es igual a numero. La distancia será evaluada en las diagonales, horizontales y verticales, si se pregunta por la distancia de dos fichas que no se encuentran en estas posiciones no será valido el predicado.

- **caminoLibre(ficha, ficha)** Es valido cuando entre las dos fichas no se interponga ninguna otra, ya sea en posición vertical, horizontal o en ambas diagonales.

Representación de caracteres

"∃" representación: "#".

"∀" representación: "@".

"," representación: ",".

"(" representación: "(".

")" representación: ")".

"not" representación: "!".

"and" representación: "&".

"or" representación: "|".

"implica" representación: ">".

Tipo de variables: [w-z][0-9]?

Ejemplos: w, w0, w1, ... , w9, x, x0, ..., y, ..., z, ..., z9.

Tipo de Constantes (que representan fichas particulares): [ptracb][0-9]?

Igual que las variables, se pueden utilizar las letras p, t, r, a, c, b al comienzo y un numero cualquiera del 0 al 9 (que es lo que indica el signo de pregunta).

Números: [0-7]

No puede haber un numero mas alto que 7 dado que excede el tablero a la hora de usarlo dentro de getFicha(numero, numero), y también lo excede a la hora de preguntar por una distancia 8.

Dadas estas condiciones el BNF que ideamos es el siguiente:

<Sentencia> ::= <Sentencia Atomica>

| <Sentencia Atomica> <Conector> <Sentencia>

| <Cuantificador> <Variable> <Sentencia>

| !<Sentencia> | (<Sentencia>)

<Sentencia Atomica> ::= <Pred_unario> (<Ficha>)

| <Pred_binario> (<Ficha> , <Ficha>)

| <Pred_dist> (<Ficha> , <Ficha> , <Numero>)

| <Pred_vacio> (<Numero> , <Numero>)

<Ficha> ::= <Funcion>(<Numero> , <Numero>) | <Constante> | <Variable>

<Numero> ::= <Variable> | <Numero_term>

<Conector> ::= & | | | >

<Cuantificador> ::= # | @

<Constante> ::= a | a0 | a1 | ... | b | b0 | ... | c | ... | p | ... | r | ... | t | ... | t9

<Numero_term> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7

<Variable> ::= w | w0 | w1 | ... | x | ... | y | ... | z | ... | z9 |

**<Pred_unario> ::= esPeon | esAlfil | esTorre | esCaballo | esReina | esRey | esBlanca
| esNegra**

**<Pred_binario> ::= mismaFila | mismaColumna | mismaDiagonal | estaEnL |
caminoLibre**

<Pred_dist> ::= distancia

<Pred_vacio> ::= estaVacio

<Funcion> ::= getFicha