Universidad de los Andes
Facultad de Ingeniería

Ingeniería de Sistemas y Computación
ISIS-1106-02 Lenguajes y Máquinas
Quiz Taller 1

February 22, 2024

Nombre: **Carné:**

Individual WORK

Modify the Prolog parser so that

1. A Functor's argument can be: variables, numbers, functors, lists, or expressions, as the ones seen in the parser tester. (exprParser/ExprParser2.jj)
2. To the right of symbol ":-" (for rules) in addition to functors, the following are also accepted:
   - Symbol "!" (this symbol may appear at most once).
   - Command `fail`. Fail may only appear before the period.
   - Command `call(A)` : A may be a variable or a functor; it cannot be a number.
   - A command of the form: `F =.. L`. where F may be a functor or a variable and L may be a variable or a list.
   - A command of the form `X := Exp`, where `X` is a variable or a number and Exp is an arithmetic expression as the ones seen in the parser tester. (exprParser/ExprParser2.jj)
   - A command of the form `A = B`, where A and B are arguments as described in (1).
   - A command of the form `A == B`, where A and B are arguments as described in (1).

For example.

```
foo(Fun,Args)  :-  C =.. [Fun | Args],  call(C).
notMember(X,[X|L]) :-  !,fail.
notMember(X,[Y|L]) :-  notMember(X,L).
notMember(X,[]).
foo(Fun2, F, A, L)  :-    F =.. [foo, A, B], Fun2 =.. L.
fact(0,1).
fact(N,F) :- N1 is N-1, fact(N1,F1), F is N*F1.
find(X,[T|Ts]) :- X is 3*T, !, find(X, Ts).
goo(X,Y,Z) :-  X1 is X+Y, X2 is X*Y, Z=[X1,Y2], X1==5.
```

Nombre _____ Código: _____