

UNIVERSIDAD NACIONAL DE SAN CRISTÓBAL DE HUAMANGA
FACULTAD DE INGENIERÍA DE MINAS, GEOLOGÍA Y CIVIL
ESCUELA PROFESIONAL INGENIERÍA DE SISTEMAS



Informe 01

CLAUSULAS DE PROLOG: HECHOS Y REGLAS

DOCENTE : Ing. QUISPE ESPILLCO, Carmen Mercedes
ASIGNATURA : Sistemas Expertos (IS-442).
ALUMNO :
PARIONA VILCA, Jhon Wilder

2019
AYACUCHO-PERÚ

Contenido

Ejercicio 01	1
Progenitores	1
1.1. Construya su árbol genealógico. Implemente las reglas que le permitan conocer los grados de parentesco: tío, tía, hermano, hermana, primo, prima, sobrino, sobrina. 1	
Ejercicio 02	4
Huracanes	4
1.2. Escriba un programa que determine a la Categoría que pertenece un huracán teniendo en cuenta la velocidad de sus vientos	4
EJERCICIO 03	7
GUSTOS	7
1.3. Escriba una regla que permita determinar el regalo perfecto para alguien, según aquello que más le gusta: regalo_perfecto(X,Y)	7
Ejercicio 04	8
Diagnostico de enfermedades	8
1.4. Agregue mayor información a la Base de conocimientos y haga las consultas correspondientes	8

EJERCICIO 01

PROGENITORES

1.1.CONSTRUYA SU ÁRBOL GENEALÓGICO. IMPLEMENTE LAS REGLAS QUE LE PERMITAN CONOCER LOS GRADOS DE PARENTESCO: TÍO, TÍA, HERMANO, HERMANA, PRIMO, PRIMA, SOBRINO, SOBRINA.



% Relaciones

```
progenitor(emilio, zenaida).
progenitor(emilio, brigida).
progenitor(emilio, pedro).
progenitor(emilio, claudia).
progenitor(teresa, zenaida).
progenitor(teresa, brigida).
progenitor(teresa, pedro).
progenitor(teresa, claudia).
progenitor(faustina, victor).
progenitor(faustina, gloria).
progenitor(gloria, angly).
progenitor(zenaida, jhon).
progenitor(zenaida, flor).
progenitor(zenaida, kevin).
progenitor(victor, jhon).
progenitor(victor, flor).
progenitor(victor, kevin).
progenitor(brigida, yonatan).
progenitor(brigida, efrain).
progenitor(brigida, liliana).
progenitor(brigida, fidelina).
progenitor(mario, yonatan).
progenitor(mario, efrain).
progenitor(mario, liliana).
progenitor(mario, fidelina).
progenitor(pedro, anghelo).
progenitor(pedro, jhoselyn).
progenitor(pedro, nayeli).
progenitor(marisol, anghelo).
progenitor(marisol, jhoselyn).
progenitor(marisol, nayeli).
progenitor(jhoselyn, yaren).
progenitor(luis, yaren).
progenitor(claudia, elvis).
progenitor(elvis, mateo).
progenitor(karen, mateo).
```



%hechos

```
mujer(faustina).
mujer(gloria).
mujer(angy).
mujer(karen).
mujer(teresa).
mujer(zenaida).
mujer(brigida).
mujer(claudia).
mujer(flor).
mujer(liliana).
mujer(jhoselyn).
mujer(marisol).
mujer(fidelina).
mujer(nayeli).

varon(emilio).
varon(mario).
varon(victor).
varon(carlos).
varon(pedro).
varon(jhon).
varon(kevin).
varon(yonatan).
varon(efrain).
varon(anghelo).
varon(elvis).
varon(yaren).
varon(mateo).
```

```

%reglas
%es madre si es progenitor o si es conyuge del progenitor
madre(MA,H):- mujer(MA),progenitor(MA,H).
padre(PA,H):- varon(PA),progenitor(PA,H).

abuelo(A,N):- padre(A,H),progenitor(H,N).
abuela(A,N):- madre(A,H),progenitor(H,N).

hermano(H,Y):- varon(H), madre(M,H), madre(M,Y), not(H==Y).
hermana(H,Y):- mujer(H), madre(M,H), madre(M,Y), not(H==Y).

primo(P,X):- varon(P), progenitor(A,P), progenitor(B,X),(hermano(A,B);hermana(A,B)).
prima(P,X):- mujer(P), progenitor(A,P), progenitor(B,X),(hermano(A,B);hermana(A,B)).

tio(T,A):- varon(T), progenitor(P,A), (hermano(T,P);primo(T,P)).
tia(T,A):- mujer(T), progenitor(P,A), hermana(T,P).

sobrino(S,T):- varon(S), progenitor(A,S), ((hermano(T,A);hermana(T,A);(primo(T,A);prima(T,A)))).
sobrina(S,T):- mujer(S), progenitor(A,S), ((hermano(T,A);hermana(T,A);(primo(T,A);prima(T,A)))).

```

Pruebas:

The screenshot shows a Prolog IDE with two files: `lab02_arbol_genealogico.pl` and `lab02_promedio.pl`. The `lab02_arbol_genealogico.pl` file contains a database of relationships. The `lab02_promedio.pl` file contains test queries. The terminal shows the results of these queries.

```

1 % Relaciones
2 progenitor(emilio, zenaida).
3 progenitor(emilio, brigida).
4 progenitor(emilio, pedro).
5 progenitor(emilio, claudia).
6 progenitor(teresa, zenaida).
7 progenitor(teresa, brigida).
8 progenitor(teresa, pedro).
9 progenitor(teresa, claudia).

?- madre(zenaida,X).
X = jhon ;
X = flor ;
X = kevin.

?- padre(mario,X).
X = yonatan ;
X = efrain ;
X = liliana ;
X = fidelina.

?- hermano(pedro,X).
X = zenaida ;
X = brigida ;
X = claudia ;
false.

?- hermana(brigida,X).
X = zenaida ;
X = pedro ;
X = claudia ;
false.

```

The image shows a Prolog IDE with two open files: `lab02_arbol_genealogico.pl` and `lab02_promedio.pl`. The `lab02_arbol_genealogico.pl` file contains the following Prolog code:

```
85 abuelo(A,N):- padre(A,H),progenitor(H,N).
86 abuela(A,N):- madre(A,H),progenitor(H,N).
87
88 hermano(H,Y):- varon(H), madre(M,H), madre(M,Y), not(H==Y).
89 hermana(H,Y):- mujer(H), madre(M,H), madre(M,Y), not(H==Y).
90
91 primo(P,X):- varon(P), progenitor(A,P), progenitor(B,X),(hermano(A,B);hermana(A,B)).
92 prima(P,X):- mujer(P), progenitor(A,P), progenitor(B,X),(hermano(A,B);hermana(A,B)).
93
94 tio(T,A):- varon(T), progenitor(P,A), (hermano(T,P);primo(T,P)).
95 tia(T,A):- mujer(T), progenitor(P,A), hermana(T,P).
```

The terminal window shows the following queries and results:

```
?- ['/home/point/Escritorio/lab02 expertos hechos reglas/lab02 arbol genealogico.pl'].
true.

?- primo(X,jhon).
X = yonatan ;
X = efrain ;
X = anghelo ;
X = elvis ;
false.

?- prima(X,efrain).
X = flor ;
X = jhoselyn ;
X = nayeli ;
false.

?- sobrino(X,kevin).
X = yaren ;
X = mateo ;
false.
```

The status bar at the bottom indicates the current position is Ln 95, Col 40, with 4 spaces, UTF-8 encoding, and LF line endings. The editor is running Prolog.

EJERCICIO 02

HURACANES

1.2.ESCRIBA UN PROGRAMA QUE DETERMINE A LA CATEGORÍA QUE PERTENECE UN HURACÁN TENIENDO EN CUENTA LA VELOCIDAD DE SUS VIENTOS

hechos:

```
%escalas de las categorias
escala(118,153,'categoria 1').
escala(154,177,'categoria 2').
escala(178,209,'categoria 3').
escala(210,250,'categoria 4').
escala(250,1000,'categoria 5').
```

```
%velocidad_huracan de los velocidad_huracanes

velocidad_huracan(agnes,140).
velocidad_huracan(franklin,140).
velocidad_huracan(barbara,130).
velocidad_huracan(dalila,130).
velocidad_huracan(cosme,140).
velocidad_huracan(erick,130).

velocidad_huracan(henriette,165).

velocidad_huracan(nora,180).
velocidad_huracan(raymond,205).

velocidad_huracan(iselle, 234).
velocidad_huracan(winnie, 232).
velocidad_huracan(harvey,2015).

velocidad_huracan(polo,321).
velocidad_huracan(allen,305).
velocidad_huracan(andrew,285).
velocidad_huracan(dean,260).
```

```

nivel_daños('categoria 1','Sin daños en las estructuras de los edificios. Daños básicamente en casas flotantes no
amarradas, arbustos y árboles. Inundaciones en zonas costeras y daños de poco alcance en puertos.').

nivel_daños('categoria 2','Daños en tejados, puertas y ventanas. Importantes daños en la vegetación, casas
móviles, etc. Inundaciones en puertos así como ruptura de pequeños amarres.').

nivel_daños('categoria 3','Daños estructurales en edificios pequeños. Destrucción de casas móviles. Las
inundaciones destruyen edificaciones pequeñas en zonas costeras y objetos a la deriva pueden causar daños en
edificios mayores. Posibilidad de inundaciones tierra adentro.').

nivel_daños('categoria 4','Daños generalizados en estructuras protectoras, desplome de tejados en edificios
pequeños. Alta erosión de bancales y playas. Inundaciones en terrenos interiores.').

nivel_daños('categoria 5','Destrucción de tejados completa en algunos edificios. Las inundaciones pueden llegar a
las plantas bajas de los edificios cercanos a la costa. Puede ser requerida la evacuación masiva de áreas
residenciales. Los vientos de 300 kmh pueden arrancar árboles y hasta casas de sus cimientos, huracán muy
poderoso.').

```

Reglas:

%reglas

```

resultado(H, C, V, MSJ):- velocidad_huracan(H,V),escala(MIN, MAX, C), MIN=<=V, V=<=MAX,
nivel_daños(C,MSJ).

```

Pruebas:

```
lab02_huracanes.pl - lab02_expertos_hechos_reglas - Visual Studio Code
lab02_arbol_genealogico.pl lab02_huracanes.pl lab02_promedio.pl
lab02_huracanes.pl
31 %nivel de daño de huracanes

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 1: Prolog
?- ['/home/point/Escritorio/lab02_expertos_hechos_reglas/lab02_huracanes.pl'].
true.

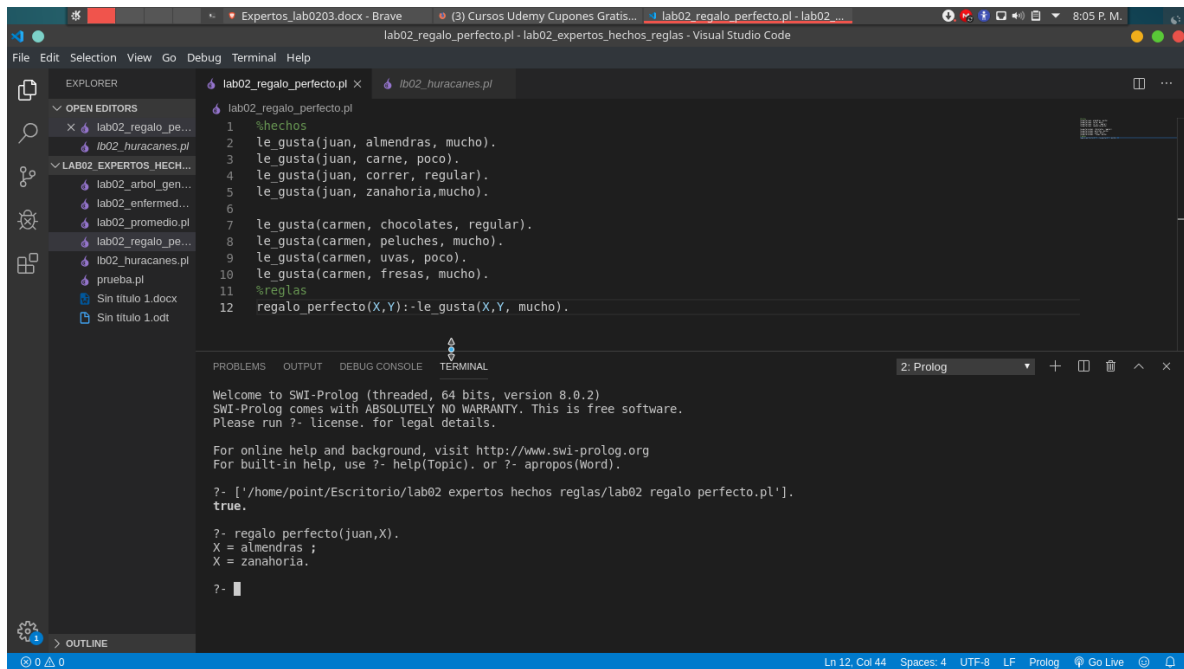
?- resultado(iselle,C,V,M).
C = 'categoria 4',
V = 234,
M = 'Daños generalizados en estructuras protectoras, desplome de tejados en edificios pequeños. Alta erosión de bancales y playas. Inundaciones en terrenos interiores.' ;
false.

?- resultado(H,'categoria 1',V,M).
H = agnes,
V = 140,
M = 'Sin daños en las estructuras de los edificios. Daños básicamente en casas flotantes no amarradas, arbustos y árboles. Inundaciones en zonas costeras y daños de poco alcance en puertos.' ;
H = franklin,
V = 140,
M = 'Sin daños en las estructuras de los edificios. Daños básicamente en casas flotantes no amarradas, arbustos y árboles. Inundaciones en zonas costeras y daños de poco alcance en puertos.' ;
H = barbara,
V = 130,
M = 'Sin daños en las estructuras de los edificios. Daños básicamente en casas flotantes no amarradas, arbustos y árboles. Inundaciones en zonas costeras y daños de poco alcance en puertos.' ;
H = dalila,
V = 130,
M = 'Sin daños en las estructuras de los edificios. Daños básicamente en casas flotantes no amarradas, arbustos y árboles. Inundaciones en zonas costeras y daños de poco alcance en puertos.' ;
H = cosme,
V = 140,
M = 'Sin daños en las estructuras de los edificios. Daños básicamente en casas flotantes no amarradas, arbustos y árboles. Inundaciones en zonas costeras y daños de poco alcance en puertos.' ;
H = erick,
V = 130,
M = 'Sin daños en las estructuras de los edificios. Daños básicamente en casas flotantes no amarradas, arbustos y árboles. Inundaciones en zonas costeras y daños de poco alcance en puertos.' ;
```


EJERCICIO 03

GUSTOS

1.3.ESCRIBA UNA REGLA QUE PERMITA DETERMINAR EL REGALO PERFECTO PARA ALGUIEN, SEGÚN AQUELLO QUE MÁS LE GUSTA:
REGALO_PERFECTO(X,Y)



```
lab02_regalo_perfecto.pl
1 %hechos
2 le.gusta(juan, almendras, mucho).
3 le.gusta(juan, carne, poco).
4 le.gusta(juan, correr, regular).
5 le.gusta(juan, zanahoria, mucho).
6
7 le.gusta(carmen, chocolates, regular).
8 le.gusta(carmen, peluches, mucho).
9 le.gusta(carmen, uvas, poco).
10 le.gusta(carmen, fresas, mucho).
11 %reglas
12 regalo_perfecto(X,Y):-le.gusta(X,Y, mucho).
```

```
Welcome to SWI-Prolog (threaded, 64 bits, version 8.0.2)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit http://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- ['/home/point/Escritorio/lab02 expertos hechos reglas/lab02 regalo perfecto.pl'].
true.

?- regalo_perfecto(juan,X).
X = almendras ;
X = zanahoria.

?-
```

EJERCICIO 04

DIAGNOSTICO DE ENFERMEDADES

1.4. AGREGUE MAYOR INFORMACIÓN A LA BASE DE CONOCIMIENTOS Y HAGA LAS CONSULTAS CORRESPONDIENTES

Hechos

```
%hechos
enfermo_de(manuel,gripe).
enfermo_de(julia,gripe).
enfermo_de(kevin,gastritis).

sintoma_de(fiebre,gripe).
sintoma_de(tos,gripe).
sintoma_de(cansancio,anemia).
sintoma_de(indigestion,gastritis).

tiene_sintoma(alicia,cansancio).
tiene_sintoma(andres,indigestion).

elimina(vitaminas,cansancio).
elimina(aspirinas,fiebre).
elimina(jarabe,tos).
elimina(antiacido,gastritis).
```

```
%reglas
recetar_a(X,Y):-enfermo_de(Y,A),alivia(X,A).

alivia(X,Y):-elimina(X,A),sintoma_de(A,Y).

enfermo_de(X,Y):-tiene_sintoma(X,Z),sintoma_de(Z,Y).
```

Pruebas:

The screenshot shows the Visual Studio Code interface with the following components:

- Explorer:** Lists files in the project, including `lab02_regalo_perfecto.pl`, `lab02_enfermedades.pl`, and several other Prolog files.
- Editor:** Displays the content of `lab02_enfermedades.pl`, which contains the following Prolog code:

```
1. enfermo_de(manuel,gripe).  
2. enfermo_de(julia,gripe).  
3. enfermo_de(kevin,gastritis).
```
- Terminal:** Shows the execution output of the Prolog program. It starts with a warning about overlapping clauses and then displays the results of several queries:

```
?- [\'/home/point/Escritorio/lab02 expertos hechos reglas/lab02 enfermedades.pl\'].  
Warning: /home/point/Escritorio/lab02 expertos hechos reglas/lab02 enfermedades.pl:25:  
Clauses of enfermo de/2 are not together in the source file  
Earlier definition at /home/point/Escritorio/lab02 expertos hechos reglas/lab02 enfermedades.pl:2  
Current predicate: alivia/2  
Use :- discontinuous enfermo de/2. to suppress this message  
  
true.  
  
?- recetar a(X,manuel).  
X = aspirinas ;  
X = jarabe ;  
false.  
  
?- recetar a(X,kevin).  
X = antiacido ;  
false.  
  
?- recetar a(X,julia).  
X = aspirinas ;  
X = jarabe ;  
false.  
  
?- recetar a(X,andres).  
X = antiacido.  
  
?- recetar a(X,alicia).  
X = vitaminas ;  
false.
```