

UNIVERSIDAD POLITECNICA SALESIANA

Narcisa Araujo

Enunciado:

Se desea modelizar el conocimiento de un experto para el diagnostico de enfermedades de las plantas, basadas en las siguientes reglas.

- Las plantas requieren diferentes tipos de nutrientes para desarrollarse adecuadamente.
- Tres de los nutrientes de mayor importancia son el nitrógeno, el fósforo y el potasio.
- Una deficiencia en alguno de estos nutrientes puede producir varios síntomas.

In [1]:

```
!pip install clipspy
from clips import Environment, Symbol

environment = Environment()
```

Requirement already satisfied: clipspy in c:\users\narcisa\anaconda3\lib\site-packages (0.3.3)

Requirement already satisfied: cffi>=1.0.0 in c:\users\narcisa\anaconda3\lib\site-packages (from clipspy) (1.14.0)

Requirement already satisfied: pycparser in c:\users\narcisa\anaconda3\lib\site-packages (from cffi>=1.0.0->clipspy) (2.19)

Escribir como reglas en CLIPS las siguientes reglas heurísticas que indican qué deficiencia se está produciendo:

- Si la planta crece muy poco entonces puede tener una deficiencia de nitrógeno.
- Si la planta tiene un color amarillo pálido entonces puede tener una deficiencia de nitrógeno.
- Si las hojas tienen un color pardo rojizo entonces la planta puede tener una deficiencia de nitrógeno.
- Si la raíz de la planta tiene poco crecimiento entonces puede tener una deficiencia de fósforo.
- Una planta con tallo fusiforme puede tener una deficiencia de fósforo.
- Una planta con color púrpura puede tener una deficiencia de fósforo.
- Un retraso en la madurez de una planta puede deberse a una deficiencia de fósforo.
- Si los bordes de las hojas aparecen chamuscados, la planta puede tener una deficiencia de potasio.
- Una planta con los tallos debilitados puede tener una deficiencia de potasio.
- Una planta con semillas o frutas marchitas puede tener una deficiencia de potasio.

REGLAS

In [2]:

```
environment.load('examen.clp')
```

In [3]:

```
environment.reset()  
environment.run()
```

Out[3]:

2

In [4]:

```
for rule in environment.rules():  
    print(rule)
```

```

(defrule MAIN::nitrogeno-1
  (planta-crece-poco)
  =>
  (assert (deficiencia-nitrogeno))
  (printout t "La planta tiene deficiencia de nitrogeno" crlf))

(defrule MAIN::nitrogeno-2
  (planta-amarillo-palido)
  =>
  (assert (deficiencia-nitrogeno))
  (printout t "La planta tiene deficiencia de nitrogeno" crlf))

(defrule MAIN::nitrogeno-3
  (hojas-pardo-rojizo)
  =>
  (assert (deficiencia-nitrogeno))
  (printout t "La planta tiene deficiencia de nitrogeno" crlf))

(defrule MAIN::fosforo-1
  (raiz-poco-crecimiento)
  =>
  (assert (deficiencia-fosforo))
  (printout t "La planta tiene deficiencia de fosforo" crlf))

(defrule MAIN::fosforo-2
  (planta-tallo-fusiforme)
  =>
  (assert (deficiencia-fosforo))
  (printout t "La planta tiene deficiencia de fosforo" crlf))

(defrule MAIN::fosforo-3
  (planta-color-purpura)
  =>
  (assert (deficiencia-fosforo))
  (printout t "La planta tiene deficiencia de fosforo" crlf))

(defrule MAIN::fosforo-4
  (retraso-madurez-planta)
  =>
  (assert (deficiencia-fosforo))
  (printout t "La planta tiene deficiencia de fosforo" crlf))

(defrule MAIN::Potasio-1
  (bordes-hojas-chamuscadas)
  =>
  (assert (deficiencia-potasio))
  (printout t "La planta tiene deficiencia de potasio" crlf))

(defrule MAIN::Potasio-2
  (planta-tallos-debilitados)
  =>
  (assert (deficiencia-potasio))
  (printout t "La planta tiene deficiencia de potasio" crlf))

(defrule MAIN::Potasio-3
  (plantas-semillas-marchitas)
  (plantas-frutas-marchitas)
  =>
  (assert (deficiencia-potasio))
  (printout t "La planta tiene deficiencia de potasio" crlf))

```

In [5]:

```
for fact in environment.facts():  
    print(fact)
```

```
(initial-fact)  
(raiz-crece-poco)  
(planta-color-purpura)  
(planta-tallos-debilitados)  
(deficiencia-potasio)  
(deficiencia-fosforo)
```

- La entrada del programa debe ser una descripción de los síntomas de la planta. La salida debe indicar de qué nutriente o nutrientes se ha producido una deficiencia imprimiendo en pantalla.

In [6]:

```
sintoma = input("Ingrese el sintoma de la planta: ")  
  
fact_string_planta = '(sintoma ' + sintoma + ' )'  
fact_planta = environment.assert_string(fact_string_planta)  
  
sintoma1 = input("Ingrese el segundo sintoma de la planta: ")  
  
fact_string_planta1 = '(sintoma1 ' + sintoma1 + ' )'  
fact_planta1 = environment.assert_string(fact_string_planta1)  
  
sintoma2 = input("Ingrese el tercer sintoma de la planta: ")  
  
fact_string_planta2 = '(sintoma2 ' + sintoma2 + ' )'  
fact_planta2 = environment.assert_string(fact_string_planta2)
```

```
Ingrese el sintoma de la planta: planta-crece-poco  
Ingrese el segundo sintoma de la planta: planta-amarillo-palido  
Ingrese el tercer sintoma de la planta: hojas-pardo-rojizo
```

In [11]:

```
environment.run()  
for fact in environment.facts():  
  
    print(fact)
```

```
(initial-fact)  
(raiz-crece-poco)  
(planta-color-purpura)  
(planta-tallos-debilitados)  
(deficiencia-potasio)  
(deficiencia-fosforo)  
(sintoma planta-crece-poco)  
(sintoma1 planta-amarillo-palido)  
(sintoma2 hojas-pardo-rojizo)
```

In []:

In []: