



**UNIVERSIDAD POLITECNICA SALESIANA**

SEDE CUENCA

**CARRERA:** INGENIERIA DE SISTEMAS

**Nombre:** Bryam Gabriel Mora Lituma

**Materia:** Sistemas Expertos

## Examen de Sistemas Expertos


### 1. Importamos las librerías para Clips.

```
In [2]: 1 from clips import Environment, Symbol  
        2 environment = Environment()
```


### 2. Limpiamos nuestro base de conocimiento.

```
In [3]: 1 environment.clear()
```

### 3. Cargamos nuestra base de Conocimiento.

In [5]:  1 environment.load('Examen.clp')

#### 4. Reseteamos y corremos nuestra base de conocimiento.

In [7]:  1 environment.reset()  
2 environment.run()

Out[7]: 0

#### 5. Recorremos nuestras Reglas.

In [9]:



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

```
(defrule MAIN::nitrogeno1  
  (La planta crece muy poco)  
  =>  
  (assert (La planta crece muy poco)))  
  
(defrule MAIN::nitrogeno2  
  (La planta tiene un color amarillo palido)  
  =>  
  (assert (La planta tiene un color amarillo palido)))  
  
(defrule MAIN::nitrogeno3  
  (La planta tiene hojas de color pardo rojizo)  
  =>  
  (assert (La planta tiene hojas de color pardo rojizo)))  
  
(defrule MAIN::fosforo1  
  (La raiz de la planata crece poco)  
  =>  
  (assert (La raiz de la planata crece poco)))  
  
(defrule MAIN::fosforo2  
  (La planta tiene tallo fusiforme)  
  =>  
  (assert (La planta tiene tallo fusiforme)))  
  
(defrule MAIN::fosforo3  
  (La planta tiene color purpura)  
  =>  
  (assert (La planta tiene color purpura)))  
  
(defrule MAIN::fosforo4  
  (La planta tiene retraso de madurez)  
  =>  
  (assert (La planta tiene retraso de madurez)))  
  
(defrule MAIN::potasio1  
  (Tiene los bordes de las hojas chamuscados)  
  =>  
  (assert (Tiene los bordes de las hojas chamuscados)))
```

```
(defrule MAIN::potasio2
  (La planta tiene tallos debilitados)
  =>
  (assert (La planta tiene tallos debilitados)))

(defrule MAIN::potasio3
  (La planta tiene semillas o frutas marchitas)
  =>
  (assert (La planta tiene semillas o frutas marchitas)))

(defrule MAIN::ausencia-nitrogeno
  (or (La planta crece muy poco)
       (La planta tiene un color amarillo palido)
       (La planta tiene hojas de color pardo rojizo))
  =>
  (assert (deficiencia-nitrogeno))
  (printout t "La planta tiene deficiencia de Nitrógeno." crlf))

(defrule MAIN::ausencia-fosforo
  (or (La raiz de la planata crece poco)
       (La planta tiene tallo fusiforme)
       (La planta tiene color purpura)
       (La planta tiene retraso de madurez))
  =>
  (assert (deficiencia-fosforo))
  (printout t "La planta tiene deficiencia de Fosforo." crlf))

(defrule MAIN::ausencia-potasio
  (or (Tiene los bordes de las hojas chamuscados)
       (La planta tiene tallos debilitados)
       (La planta tiene semillas o frutas marchitas))
  =>
  (assert (deficiencia-potasio))
  (printout t "La planta tiene deficiencia de Potasio." crlf))
```

## 6. Ingresamos los problemas de la planta.

```
In [10]: 1 environment.reset()
2 problema1 = input("Que problema tiene su planta ? ")
3 fact_string_planta1 = '(' + problema1 + ')'
4 fact_planta1 = environment.assert_string(fact_string_planta1)
5
6
7 problema2 = input("Que problema tiene su planta ? ")
8 fact_string_planta2 = '(' + problema2 + ')'
9 fact_planta2 = environment.assert_string(fact_string_planta2)
10
11
12 problema3 = input("Que problema tiene su planta ? ")
13 fact_string_planta3 = '(' + problema3 + ')'
14 fact_planta3 = environment.assert_string(fact_string_planta3)
```

Que problema tiene su planta ? La raiz de la planta crece muy poco  
Que problema tiene su planta ? La planta tiene color purpura  
Que problema tiene su planta ? La planta tiene tallos debilitados

## 7. Recoremos nuestros facts para ver nuestros hechos iniciales.

```
In [11]: 1 for fact in environment.facts():
2         print(fact)
```

(initial-fact)  
(La raiz de la planta crece muy poco)  
(La planta tiene color purpura)  
(La planta tiene tallos debilitados)

## 8. Coremos nuestra base de conocimiento y mediante nuestros hechos iniciales nos da las deficiencias de la planta.

```
In [13]: 1 environment.run()
          2
          3 for fact in environment.facts():
          4     if fact.template.name == 'deficiencia-nitrogeno':
          5         print('La planta tiene deficiencia de Nitrógeno')
          6     elif fact.template.name == 'deficiencia-fosforo':
          7         print('La planta tiene deficiencia de Fósforo')
          8     elif fact.template.name == 'deficiencia-potasio':
          9         print('La planta tiene deficiencia de Potasio')
```

La planta tiene deficiencia de Potasio

La planta tiene deficiencia de Fósforo

```
In [ ]: 1
```