

**UNIVERSIDAD POLITECNICA SALESIANA**

**SEMAFORO clips**

**Fabian Armijos**

```
(defrule tipo-robot-regla
(initial-fact)
=>
(printout t "INGRESE TIPO ROBOT 1 (peaton/carro)? " crlf)
(assert (tipo-robot (read)))
)
```

```
(defrule semaforo-tipo
(or (tipo-robot peaton) (tipo-robot carro))
=>
(printout t "Ingrese color del semaforo" crlf)
(printout t "rojo/verde/amarillo" crlf)
(assert (semaforo (read)))
)
```

```
(defrule pasar
(tipo-robot ?tipo)
(or (and (tipo-robot carro) (semaforo verde)) (and (tipo-robot peaton) (semaforo rojo))) =>
(printout t "El robot " ?tipo " puede pasar" crlf)
)
```

```
(defrule precaucion
(tipo-robot ?tipo)
(semaforo amarillo)
=>
(printout t "El robot " ?tipo " pasar con precaucion (corre !!!)" crlf)
)
```

```

(defrule esperar
(tipo-robot ?tipo)

(or (and (tipo-robot carro) (semaforo rojo)) (and (tipo-robot peaton) (semaforo verde)))) =>

(printout t "El robot " ?tipo " no puede pasar esperar" crlf)
)

```

## Ejecución

```

CLIPS (6.31 6/12/19)
CLIPS> (load "semaforo.clp")
Defining defrule: tipo-robot-regla +j+j
Defining defrule: semaforo-tipo +j+j
+j+j
Defining defrule: pasar +j+j+j+j
=j+j+j+j
Defining defrule: precaucion =j+j+j
Defining defrule: esperar =j=j+j+j
=j=j+j+j
TRUE

```

```

INGRESE TIPO ROBOT 1 (peaton/carro)?
carro
Ingrese color del semaforo
rojo/verde/amarillo
rojo
El robot carro no puede pasar debe esperar
CLIPS>

```

```

INGRESE TIPO ROBOT 1 (peaton/carro)?
carro
Ingrese color del semaforo
rojo/verde/amarillo
verde
El robot carro puede pasar
CLIPS>

```

```

INGRESE TIPO ROBOT 1 (peaton/carro)?
peaton
Ingrese color del semaforo
rojo/verde/amarillo
rojo
El robot peaton puede pasar
CLIPS>

```

```
INGRESE TIPO ROBOT 1 (peaton/carro)?  
peaton  
Ingrese color del semaforo  
rojo/verde/amarillo  
verde  
El robot peaton no puede pasar debe esperar  
CLIPS> █
```