


```
(defrule tipo-robot-regla

(initial-fact)

=>

(printout t "QUE TIPO DE ROBOT/CONFIGURACION TIENES (peaton/carro)?" crlf)

(assert (tipo-robot (read)))

)

(defrule semaforo-color-regla

(or (tipo-robot peaton) (tipo-robot carro))

=>

(printout t "Ingrese el color del semaforo (rojo/verde/amarillo)?" crlf)

(assert (semaforo (read)))

)

(defrule pasar-regla

(tipo-robot ?tipo)

(semaforo verde)

=>

(printout t "El robot: " ?tipo " puede pasar " crlf)

)

(defrule precaucion-regla

(tipo-robot ?tipo)

(semaforo amarillo)

=>

(printout t "El robot: " ?tipo " pasar rapido " crlf)

)

(defrule detener-regla

(tipo-robot ?tipo)

(semaforo rojo)

=>
```

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

)

MODIFICAR LA REGLA DEL SEMAFORO: MODIFICAR EL SISTEMA PARA QUE SOLO EXSTA UN SEMAFORO Y QUE SEGÚN EL COLOR IMPRIMA QUE PUEDE HACER EL ROBOT

```
(defrule tipo-robot-regla
(initial-fact)
=>
(printout t "QUE TIPO DE ROBOT/CONFIGURACION TIENES (peaton/carro)?" crlf)
(assert (tipo-robot (read)))
)
(defrule semaforo-color-regla
(or (tipo-robot peaton) (tipo-robot carro))
=>
(printout t "Ingrese el color del semaforo (rojo/verde/amarillo)?" crlf)
(assert (semaforo (read)))
)
(defrule general-regla
(tipo-robot ?tipo)
(semaforo ?color)
=>
(if (eq ?color verde)
then
(printout t "El robot: " ?tipo " puede pasar " crlf)
)
(if (eq ?color amarillo)
then
(printout t "El robot: " ?tipo " pasar rapido " crlf)
)
(if (eq ?color rojo)
then
(printout t "El robot: " ?tipo " no puede pasar " crlf)
)
```

)