```
(defrule tipo-triangulo-regla
=>
(printout t "Tipo de triangulo (1. Lado/ 2. ├üngulo)? " crlf)
(assert (tipo-triangulo(read)))
)

(defrule tipo-lados
(tipo-triangulo 1)
=>
(printout t  "Ingrese los tres lados " crlf)
(assert (pl(read)))
(assert (sl(read)))
(assert (tl(read)))
)

(defrule tipo-angulos
(tipo-triangulo 2)
=>
(printout t  "Ingrese los tres angulos " crlf)
(assert (pa(read)))
(assert (sa(read)))
(assert (ta(read)))
)

(defrule equilatero
(pl ?x)
(sl ?y)
(tl ?z)
(test (and (= ?x ?y) (= ?x ?z)))
=>
(printout t "Es Equilatero" crlf)
)

(defrule isosceles
(pl ?x)
(sl ?y)
(tl ?z)
(test (or (and (= ?y ?z) (!= ?y ?x)) (or (and (= ?x ?y) (!= ?x ?z)) (and
(!= ?x ?y) (= ?x ?z)))))
=>
(printout t "Es Isoceles" crlf)
)

(defrule escaleno
(pl ?x)
(sl ?y)
(tl ?z)
(test (or (and (!= ?y ?z) (!= ?y ?x)) (or (and (!= ?x ?y) (!= ?x ?z)) (and
(!= ?x ?y) (!= ?x ?z)))))
=>
(printout t "Es Escaleno" crlf)
)

(defrule rectangulo
(pa ?x)
(sa ?y)
(ta ?z)
(test (and (= 180 (+ ?x (+ ?y ?z))) (or (= ?x 90) (or (= ?y 90) (= ?z
90)))))
=>
(printout t "Es Rectangulo" crlf)
)

(defrule acutangulo
(pa ?x)
(sa ?y)
(ta ?z)
(test (and (< ?x 90) (and (< ?y 90) (and (< ?z 90) (= 180 (+ ?x (+ ?y
?z)))))))
=>
(printout t "Es Acutangulo" crlf)
)

(defrule obtusangulo
(pa ?x)
(sa ?y)
(ta ?z)
(test (and (= 180 (+ ?x (+ ?y ?z)))
(or (or (and (> ?x 90) (and (< ?y 90) (< ?z 90))) (and (> ?y 90) (and (<
?x 90) (< ?z 90)))) (and (> ?z 90) (and (< ?x 90) (< ?y 90))))))
=>
(printout t "Es Obtuso" crlf)
)
```