

```
(deffacts reglas
(B) (A)
)

(defrule letra-A (A) => (assert (E)) (printout t "activo E" crlf))
(defrule letra-B (B) => (assert (D)) (printout t "activo D" crlf))
(defrule letra-H (H) => (assert (A)) (printout t "activo A" crlf))
(defrule letra-EG (E) (G) => (assert (C)) (printout t "activo C" crlf))
(defrule letra-EK (E) (K) => (assert (B)) (printout t "activo B" crlf))
(defrule letra-DEK (D) (E) (K) => (assert (C)) (printout t "activo C"
crlf))
(defrule letra-GKF (G) (K) (F) => (assert (A)) (printout t "activo A" crlf)
)
```