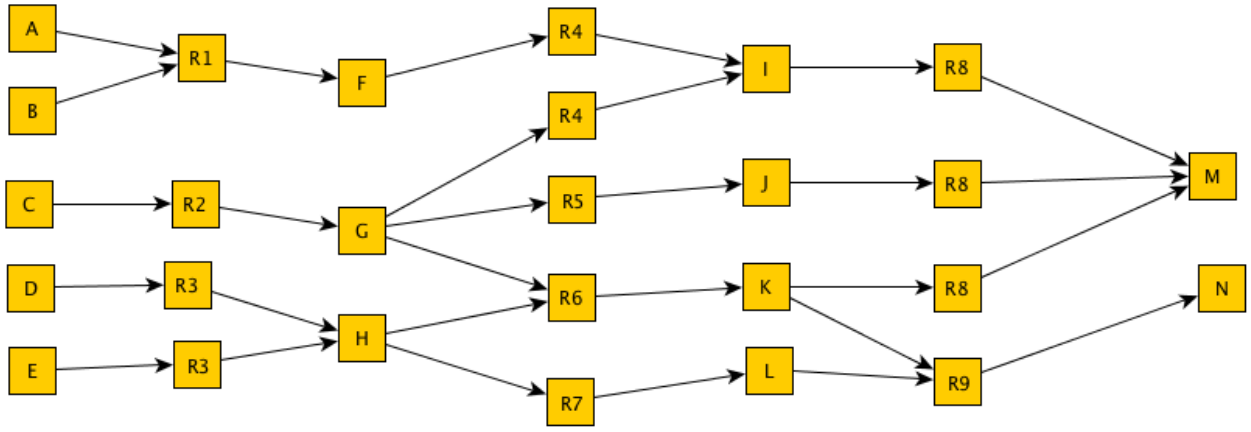
Aufgabe 13

a)



b)

```
=================== Aufgabe 13b (Vorwaertsverkettung) ===================
; Regeln fuer Vorwaertsverkettung
(watch rules)
(watch facts)
(watch activations)

; Breiten- oder Tiefensuche
;(set-strategy breadth)
(set-strategy depth)

; Regelmenge
(defrule R1 (A) (B)             => (assert (F)))
(defrule R2 (C)                         => (assert (G)))
(defrule R3 (or (D) (E))        => (assert (H)))
(defrule R4 (or (F) (G))        => (assert (I)))
(defrule R5 (G)                         => (assert (J)))
(defrule R6 (G) (H)             => (assert (K)))
(defrule R7 (H)                         => (assert (L)))
(defrule R8 (or (I) (J) (K))    => (assert (M)))
(defrule R9 (K) (L)             => (assert (N)))
(defrule R11 (declare (salience -100)) => (facts))


=========================       Aufgabe 13b, Breitensuche ========================
Jess, the Java Expert System Shell
Copyright (C) 2001 E.J. Friedman Hill and the Sandia Corporation
Jess Version 6.1p2 5/21/2003
```

```
Jess> (batch aufgabe13.clp)
TRUE
Jess> (reset)
 ==> f-0 (MAIN::initial-fact)
==> Activation: MAIN::R11 :  f-0
TRUE
Jess> (assert (A) (B) (C))
 ==> f-1 (MAIN::A)
 ==> f-2 (MAIN::B)
==> Activation: MAIN::R1 :  f-1, f-2
 ==> f-3 (MAIN::C)
==> Activation: MAIN::R2 :  f-3
<Fact-3>
Jess> (run 1)
FIRE 1 MAIN::R1 f-1, f-2
 ==> f-4 (MAIN::F)
==> Activation: MAIN::R4 :  f-4
1
Jess> (run 1)
FIRE 1 MAIN::R2 f-3
 ==> f-5 (MAIN::G)
==> Activation: MAIN::R4 :  f-5
==> Activation: MAIN::R5 :  f-5
1
Jess> (run 1)
FIRE 1 MAIN::R4 f-4
 ==> f-6 (MAIN::I)
==> Activation: MAIN::R8 :  f-6
1
Jess> (run 1)
FIRE 1 MAIN::R4 f-5
1
Jess> (run 1)
FIRE 1 MAIN::R5 f-5
 ==> f-7 (MAIN::J)
==> Activation: MAIN::R8 :  f-7
1
Jess> (run 1)
FIRE 1 MAIN::R8 f-6
 ==> f-8 (MAIN::M)
1
Jess> (run 1)
FIRE 1 MAIN::R8 f-7
1
Jess> (run 1)
FIRE 1 MAIN::R11 f-0
f-0   (MAIN::initial-fact)
f-1   (MAIN::A)
f-2   (MAIN::B)
f-3   (MAIN::C)
f-4   (MAIN::F)
f-5   (MAIN::G)
f-6   (MAIN::I)
f-7   (MAIN::J)
f-8   (MAIN::M)
For a total of 9 facts.
```

1
Jess>

========== Aufgabe 13b, Tiefensuche ==========

```
Jess, the Java Expert System Shell
Copyright (C) 2001 E.J. Friedman Hill and the Sandia Corporation
Jess Version 6.1p2 5/21/2003

Jess> (batch aufgabe13.clp)
TRUE
Jess> (reset)
 ==> f-0 (MAIN::initial-fact)
==> Activation: MAIN::R11 :  f-0
TRUE
Jess> (assert (A) (B) (C))
 ==> f-1 (MAIN::A)
 ==> f-2 (MAIN::B)
==> Activation: MAIN::R1 :  f-1, f-2
 ==> f-3 (MAIN::C)
==> Activation: MAIN::R2 :  f-3
<Fact-3>
Jess> (run 1)
FIRE 1 MAIN::R2 f-3
 ==> f-4 (MAIN::G)
==> Activation: MAIN::R4 :  f-4
==> Activation: MAIN::R5 :  f-4
1
Jess> (run 1)
FIRE 1 MAIN::R4 f-4
 ==> f-5 (MAIN::I)
==> Activation: MAIN::R8 :  f-5
1
Jess> (run 1)
FIRE 1 MAIN::R8 f-5
 ==> f-6 (MAIN::M)
1
Jess> (run 1)
FIRE 1 MAIN::R5 f-4
 ==> f-7 (MAIN::J)
==> Activation: MAIN::R8 :  f-7
1
Jess> (run 1)
FIRE 1 MAIN::R8 f-7
1
Jess> (run 1)
FIRE 1 MAIN::R1 f-1, f-2
 ==> f-8 (MAIN::F)
==> Activation: MAIN::R4 :  f-8
1
Jess> (run 1)
FIRE 1 MAIN::R4 f-8
1
Jess> (run 1)
FIRE 1 MAIN::R11 f-0
f-0   (MAIN::initial-fact)
f-1   (MAIN::A)
```

```
f-2   (MAIN::B)
f-3   (MAIN::C)
f-4   (MAIN::G)
f-5   (MAIN::I)
f-6   (MAIN::M)
f-7   (MAIN::J)
f-8   (MAIN::F)
For a total of 9 facts.
1
Jess>
```

c)

```
; Regeln fuer Vorwaertsverkettung
;(watch rules)
;(watch facts)
;(watch activations)

; Breiten- oder Tiefensuche
;(set-strategy breadth)
;(set-strategy depth)

(do-backward-chaining M)
(do-backward-chaining N)

; Regelmenge
(defrule R1 (A) (B) (need-M)                    => (assert (F)))
(defrule R2 (C) (or (need-M) (need-N))                  => (assert (G)))
(defrule R3 (or (D) (E)) (need-N)               => (assert (H)))
(defrule R4 (or (F) (G)) (need-M)               => (assert (I)))
(defrule R5 (G) (need-M)                        => (assert (J)))
(defrule R6 (G) (H) (or (need-M) (need-N))      => (assert (K)))
(defrule R7 (H) (need-N)                        => (assert (L)))
(defrule R8 (or (I) (J) (K)) (need-M)           => (assert (M)))
(defrule R9 (K) (L) (need-N)                    => (assert (N)))
(defrule R11 (declare (salience -100))                  => (facts))

(defrule TRUE_M (M) => (printout t "------> M is true <-------" crlf))
(defrule TRUE_N (N) => (printout t "------> N is true <-------" crlf))

(defrule NO_N (declare (salience -100)) (need-N) (not (N)) => (printout t "N cannot be confirmed
for given facts" crlf))
(defrule NO_M (declare (salience -100)) (need-M) (not (M)) => (printout t "M cannot be confirmed
for given facts" crlf))

================== Aufgabe 13c (Rueckwaertsverkettung) ==================
Jess, the Java Expert System Shell
Copyright (C) 2001 E.J. Friedman Hill and the Sandia Corporation
Jess Version 6.1p2 5/21/2003

Jess> (batch aufgabe13b.clp)
TRUE
Jess> (reset)
TRUE
Jess> (assert (A) (B) (C))
```

```
<Fact-5>
Jess> (run)
------> M is true <-------
N cannot be confirmed for given facts
f-0   (MAIN::initial-fact)
f-1   (MAIN::need-M)
f-2   (MAIN::need-N)
f-3   (MAIN::A)
f-4   (MAIN::B)
f-5   (MAIN::C)
f-6   (MAIN::G)
f-7   (MAIN::I)
f-8   (MAIN::M)
For a total of 9 facts.
6
Jess>
```

d)

Bei der Vorwärtsverkettung haben wir sowohl bei Breiten als auch bei Tiefensuche 8 gefeuerte Regeln, bei der Rückwärtsverkettung sind es 9. Bei der Tiefensuche werden die Regeln 2,4,8,5,8,1 in dieser Reihenfolge gefeuert und bei der Breitensuche 1,2,4,4,5,8,8,11. Von der Faktenbasis unterscheidet sich Vorwärtsverkettung und Rückwärtsverkettung um F und J, welche die Vorwärtsverkettung mehr hat, da diese Fakten zum Beweisen der Hypothesen nicht benötigt werden.