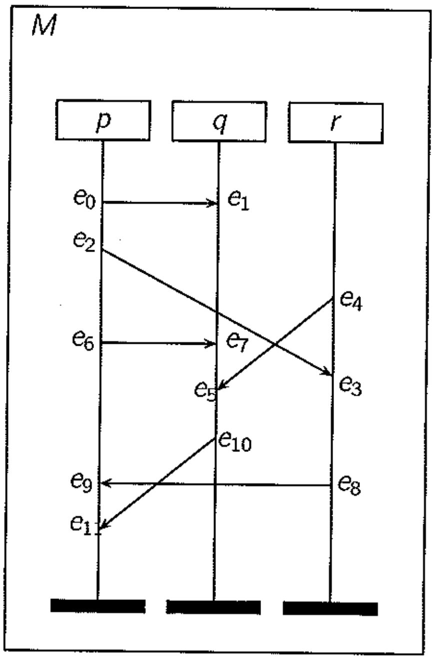
Prof. Dr. Ir. Joost-Pieter Katoen

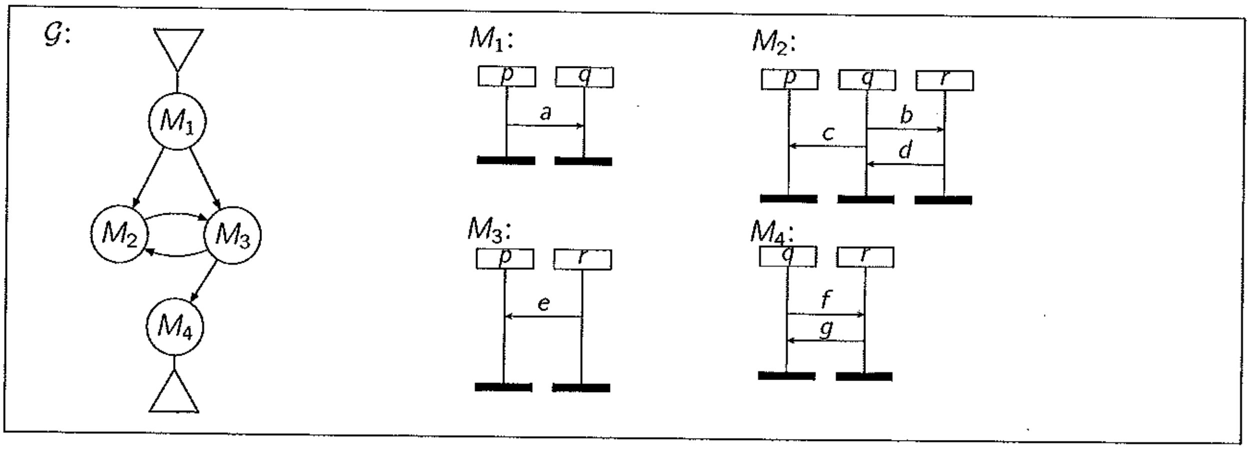
Falak Sher, Sabrina von Styp

Exercise 1 (Races): (5+5 points)

a) Does the following MSC M have a race? If yes, indicate all pairs of events that form a race.



b) Consider the following MSG $\mathcal{G}$. Is there a path so that $\mathcal{G}$ has a race?



Solution:

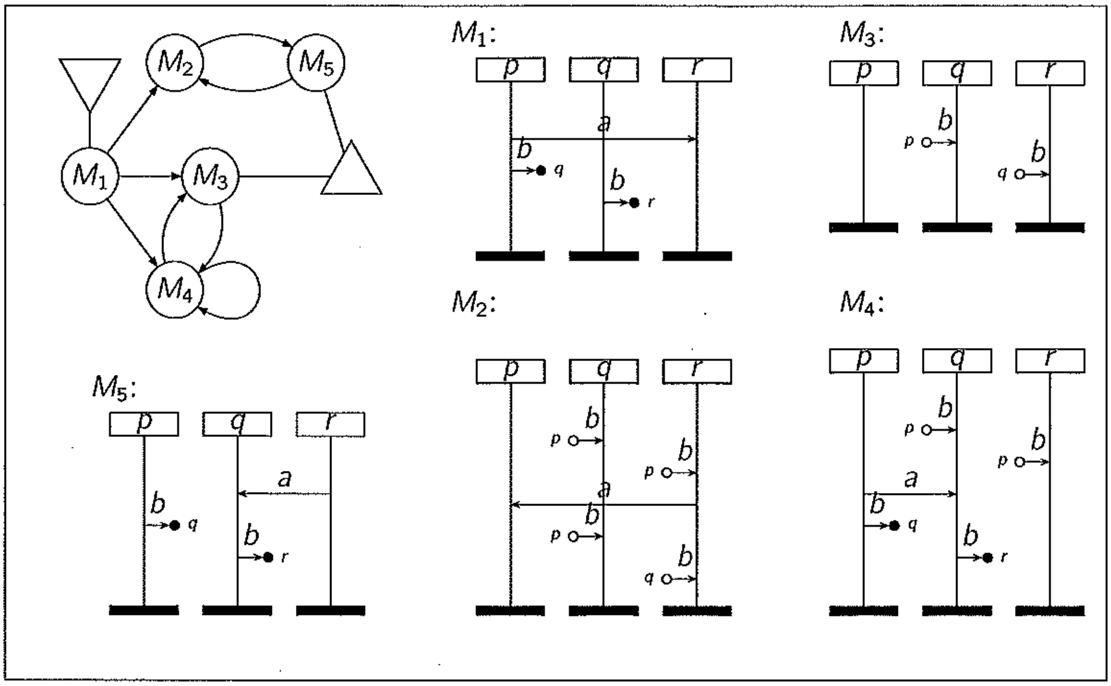
a) The MSC M_1 has the following three races:

- e_7 and e_5
- e_1 and e_5
- e_9 and e_{11}

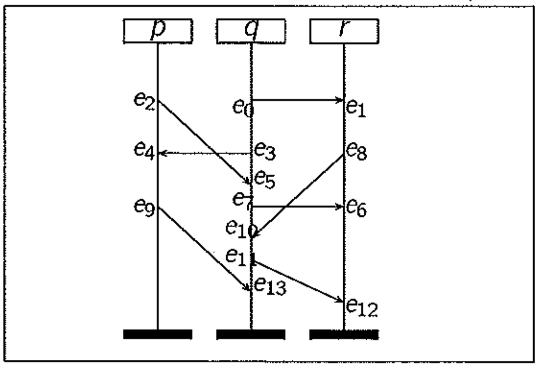
b) The MSG M has a race on the path $M_1M_2M_3M_4$. The race is between receiving message c and e in process p.

Exercise 2 (Safeness and ∃/∀ boundedness): (4+3+3 points)

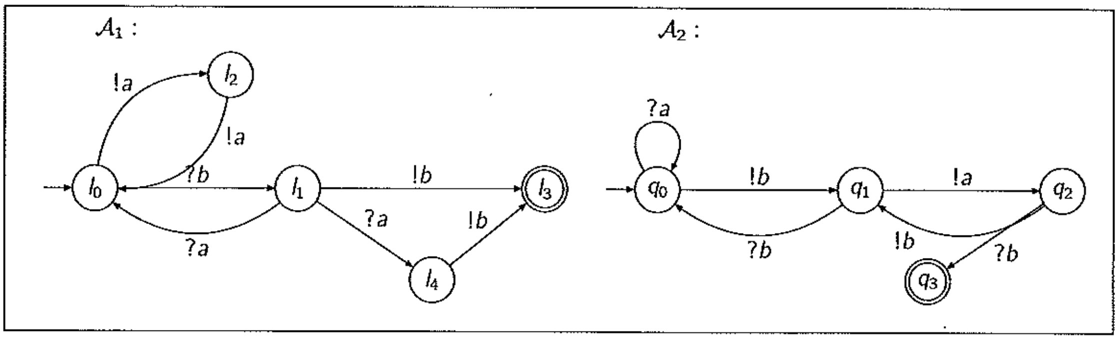
a) Determine if the following CMSG is safe. Justify your answer.



b) Determine for the following MSC if it is existentially (∃−) or universally (∀−) bounded. In case it is ∃/∀-bounded, determine the smallest B such that the MSC is ∃/∀-B-bounded and argue why it cannot be $\exists/\forall-(B-1)$−bounded.



c) Let the following CFM $\mathcal{A}$, described by $\mathcal{A}_1$ and $\mathcal{A}_2$, be given. Is the CFM $\mathcal{A}$ -∃/∀-B-bounded? (if the answer is *yes* find the smallest such B)



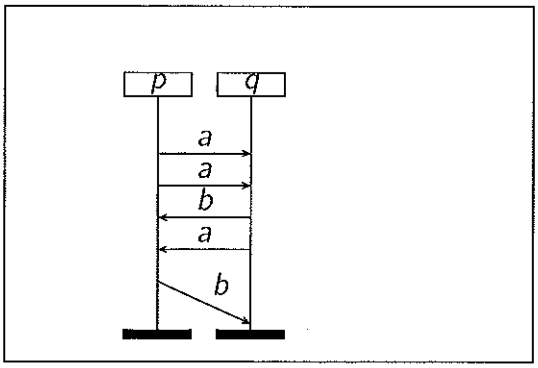
Solution:

a) The CMSG $\mathcal{G}$ is not safe the MSCs that ist generated when concatenating $M_1M_2M_5M_2M_5$ has two unmatched receive messages.

b) M is $\exists - 1-$bounded, as every message can be received before the next one for the same channel is sent.

M is $\forall - 2-$bounded, as the q can only sent two of its three messages to r e_0 and e_7 then e_1 has to happen as e_8 has to be executed since $e_{10} <_p e_{11}$.

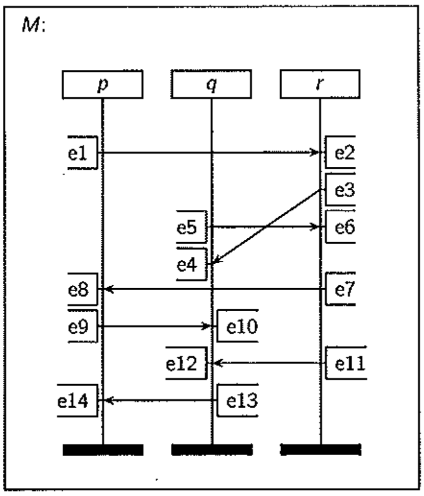
c) The DFA $\mathcal{G}$ is $\exists - 1-$bounded, as the corresponding MSC is $\exists - 1-$bounded.



The $\mathcal{G}$ is not $\forall-$bounded, as $\mathcal{A}_1$ can loop between location l_0 and l_2 and $\mathcal{A}_2$ must not receive any messages.

Exercise 3 (PDL): (10 points)

a) Consider the following MSC M defined over the set of processes $\mathcal{P} = \{p, q, r\}$.



Determine whether the MSC M satisfies the following PDL formulas or not. If your answer is 'yes', provide at least one event that satisfies the corresponding formula.

1) $\exists\langle (proc + msg)^*; \{?(p, q, \cdot)\}\rangle [proc]^{-1}\langle ((proc + msg)^*\rangle ?(r, q, \cdot)$
2) $\exists\langle \{!p\}; (proc + \{[msg]^{-1}\}; \{!q \vee ?p\})^*\rangle ?p$
3) $\exists\langle msg\rangle [proc]^{-1}\langle msg\rangle [proc; proc]^{-1} false$

Note: for $p_1, p_2 \in \mathcal{P}$, $?(p_1, p_2, \cdot)$ abbreviates $\bigvee_{a\in\mathcal{C}} ?(p_1, p_2, a)$ where $\mathcal{C}$ and $\mathcal{P}$ are the sets of message contents and processes in the MSC M respectively. Moreover, we define $?p_1 = \bigvee_{p'\in\mathcal{P}\setminus\{p_1\}, a\in\mathcal{C}} ?(p_1, p', a)$. Similarly we define $!p_1$.

b) Write down the PDL formulas that correspond to the informal descriptions about the MSC M:

1) Once process p receives a message from process r, it will not receive any message any further.
2) Every message that process r receives from process q is immediately passed from process r to process p.

Solution:

a) The satisfaction PDL formulas by MSC M is given as:

1) No. Only one event $e14$ is labelled with $?(p,q,\cdot)$. As $e14$ is the only maximal event of the MSC M, every other event has a path that eventually reaches $e14$. From $e14$, we reach $e9$ by performing $[proc]^{-1}$ and going one step back along the process p. From the event $e9$, by doing $proc$ or msg, there is no way to reach to event $e6$ that is labelled with $?(r,q,\cdot)$. Hence, no event satisfies this formula.

2) Yes, both $e1$ and $e9$ satisfy the formula.
Both $e1$ and $e9$ are sent-events at process p. By performing $proc$ from $e1$ and $e9$, we reach $e8$ and $e14$ respectively which are receive-events at process p. $e8$ and $e14$ satisfy $\{!q \vee ?p\}$ and $?p$. Hence, the formula is satisfied by $e1$ and $e9$.

3) Yes, $e5$ satisfies the formula. We present the following path that shows the satisfaction of the formula by $e5$.
$e5 \xrightarrow{msg} e6 \xrightarrow{[proc]^{-1}} e3 \xrightarrow{msg} e4 \xrightarrow{[proc]^{-1}} e5 \xrightarrow{[proc]^{-1}} false.$

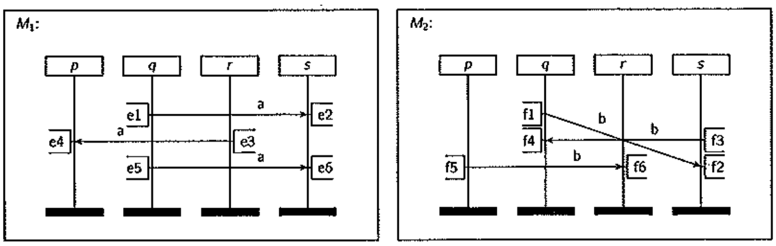
b) 1) $\forall\Big(?(p,r,\cdot) \rightarrow [(proc;\{\neg ?p\})^*; proc] false\Big)$.

2) $\forall \bigwedge_{a\in C} \Big(?(r,q,a) \rightarrow [proc]!(r,p,a)\Big)$. or
$\bigwedge_{a\in C} \forall\Big(?(r,q,a) \rightarrow [proc]!(r,p,a)\Big)$.

Note: for $p_1, p_2 \in \mathcal{P}$, $?(p_1,p_2,\cdot)$ abbreviates $\bigvee_{a\in C} ?(p_1,p_2,a)$ where $\mathcal{C}$ and $\mathcal{P}$ are the sets of message contents and processes in the MSC M respectively. Moreover, we define $?p_1 = \bigvee_{p'\in\mathcal{P}\setminus\{p_1\}, a\in C} ?(p_1,p',a)$.

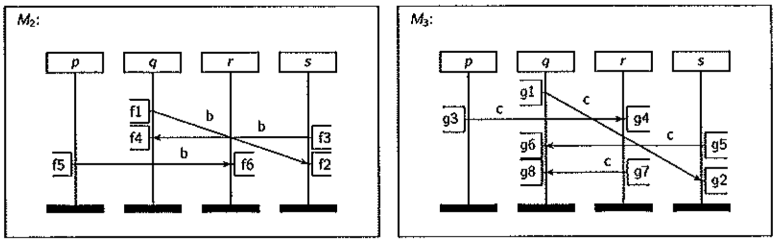
Exercise 4 (Realizability): (10 points)

a) Consider the following MSCs M_1 and M_2 defined over the set of processes $\mathcal{P} = \{p, q, r, s\}$.



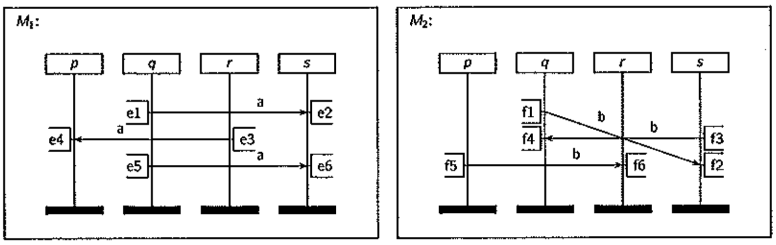
Prove that $L_1 = Lin(M_1) \cup Lin(M_2)$ is not closed under $\models$.

b) Consider the following MSCs M_2 and M_3 defined over the set of processes $\mathcal{P} = \{p, q, r, s\}$.



Prove that $L_2 = Lin(M_2) \cup Lin(M_3)$ is not closed under $\models^{df}$.

c) Consider the following MSCs M_1 and M_2 defined over the set of processes $\mathcal{P} = \{p, q, r, s\}$.



Modify either M_1 or M_2 by adding only one pair of send and receive events such that $Lin(M_1) \cup Lin(M_2)$ is closed under $\models$.

Solution:

a) Let $w = e_3e_4f_1f_3f_2f_4$ be a well-formed word, where each $l(e_i), l(f_i) \in Act$. We have projections of w on each process in $\mathcal{P} = \{p, q, r, s\}$ as:

- $w_{\upharpoonright_p} = e_4$
- $w_{\upharpoonright_q} = f_1 f_4$
- $w_{\upharpoonright_r} = e_3$
- $w_{\upharpoonright_s} = f_3 f_2$

For each process $j \in \mathcal{P} = \{p, q, r, s\}$, we find a word $v \in L_1$ such that $v_{\upharpoonright_j} = w_{\upharpoonright_j}$.

- For process p, we have $v = e_1 e_2 e_3 e_4 e_5 e_6$ and $v_{\upharpoonright_p} = e_4 = w_{\upharpoonright_p} = e_4$.
- For process q, we have, $v = f_1 f_3 f_2 f_4 f_5 f_6$ and $v_{\upharpoonright_q} = f_1 f_4 = w_{\upharpoonright_q} = f_1 f_4$.
- For process r, we have $v = e_1 e_2 e_3 e_4 e_5 e_6$ and $v_{\upharpoonright_r} = e_3 = w_{\upharpoonright_r} = e_3$.
- For process s, we have $v = f_1 f_3 f_2 f_4 f_5 f_6$ and $v_{\upharpoonright_s} = f_3 f_2 = w_{\upharpoonright_s} = f_3 f_2$.

This shows that $L_1 \models w$ but $w \notin pref(L_1)$. Hence L_1 is not closed under $\models$.

b) Let $w = f_1 g_5 g_2$ be a proper word, where each $l(f_i), l(g_i) \in Act$. We have projections of w on each process in $\mathcal{P} = \{p, q, r, s\}$ as:

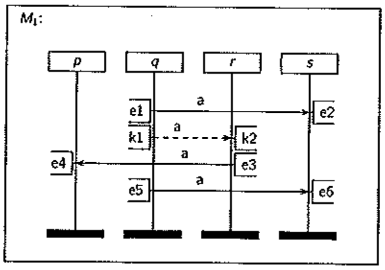
- $w_{\upharpoonright_p} = \epsilon$
- $w_{\upharpoonright_q} = f_1$
- $w_{\upharpoonright_r} = \epsilon$
- $w_{\upharpoonright_s} = g_5 g_2$

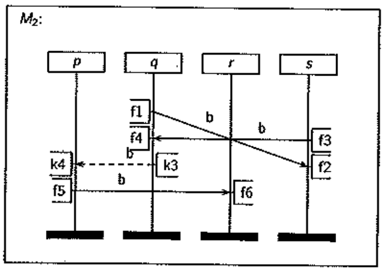
For each process $j \in \mathcal{P} = \{p, q, r, s\}$, we find a word $v \in L_2$ such that $w_{\upharpoonright_j}$ is a prefix of $v_{\upharpoonright_j}$.

- For process p, we have $v = f_1 f_3 f_2 f_4 f_5 f_6$ and $w_{\upharpoonright_p} = \epsilon$ is a prefix of $v_{\upharpoonright_p} = f_5$.
- For process q, we have $v = f_1 f_3 f_2 f_4 f_5 f_6$ and $w_{\upharpoonright_q} = f_1$ is a prefix of $v_{\upharpoonright_q} = f_1 f_4$.
- For process r, we have $v = f_1 f_3 f_2 f_4 f_5 f_6$ and $w_{\upharpoonright_r} = \epsilon$ is a prefix of $v_{\upharpoonright_r} = f_6$.
- For process s, we have $v = g_1 g_5 g_2 g_6 g_3 g_4 g_7 g_8$ and $w_{\upharpoonright_s} = g_5 g_2$ is a prefix of $v_{\upharpoonright_s} = g_5 g_2$.

This shows that $L_2 \models^{df} w$ but $w \notin pref(L_2)$. Hence L_2 is not closed under $\models^{df}$.

c) Let $L_3 = Lin(M_1) \cup Lin(M_2)$. We only need to make M_1 or M_2 weakly connected; and that can be done by adding just one pair of events in M_1 or M_2. The language L_3 will then become closed under $\models$, and thus realizable. Following could be the two possibly ways to modify MSCs.

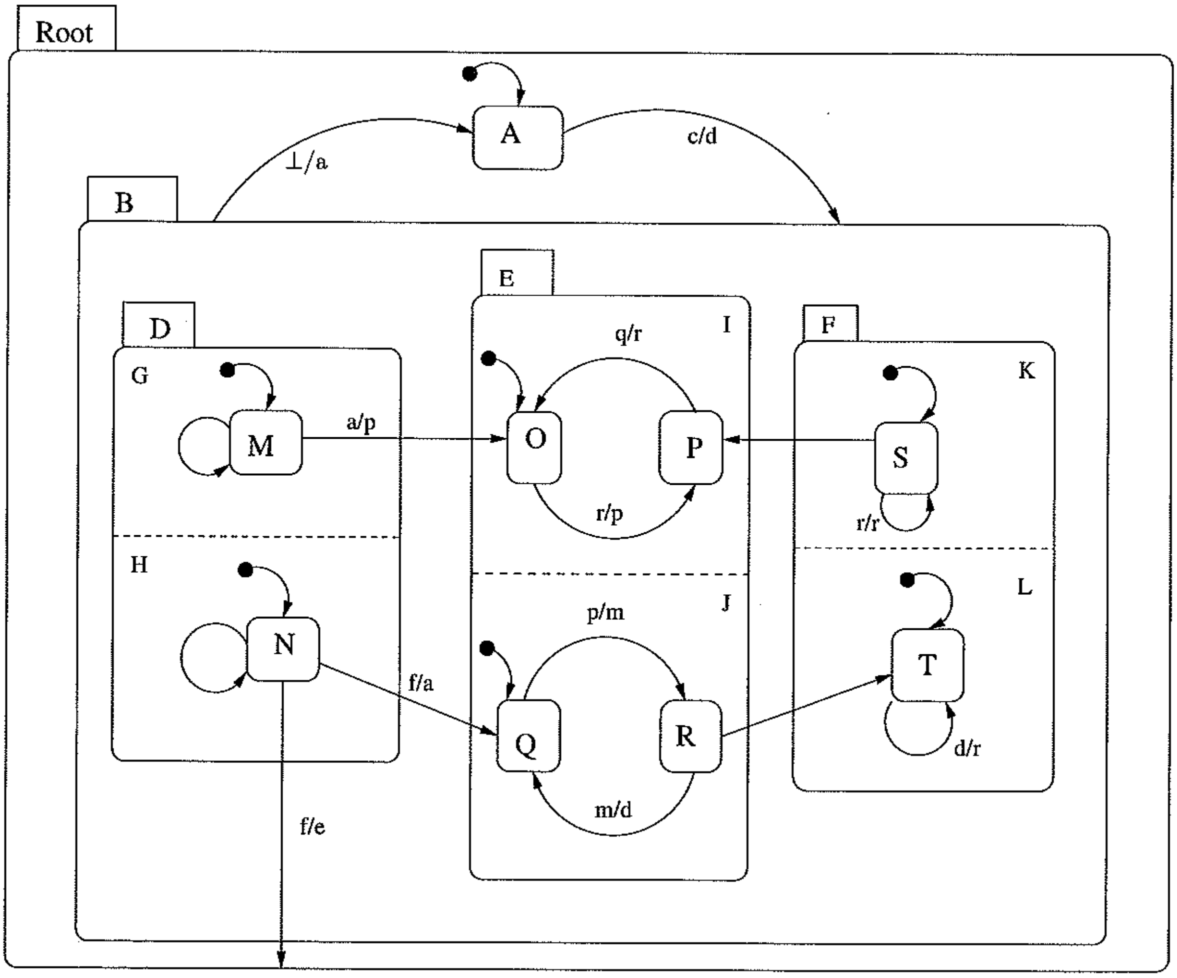




Now we are not able to find even a single well-formed word $w \in Act^*$ such that $L_3 \models w$ and $w \notin L_3$. Hence, L_3 is closed under $\models$.

Exercise 5 (Statechart): (10 points)

Let the following statechart $\mathbb{S} = (N, E, Edges)$ be given:



Note: In this assignment an edge label of the form e/e' of Statechart $\mathbb{S}$ means that $\mathbb{S}$ is consuming event e and executing an action that is sending the event e' to $\mathbb{S}$ (i.e., to itself).

a) Give the type of the nodes A, B, D and N.

b) Construct the tree that represents the node hierarchy of statechart $\mathbb{S}$.

c) Determine the priority between:

1) moving from N to Q and moving from N to $Root$, and
2) moving from M to M and moving from M to O

provided both the edges are enabled in each of the above cases.

d) Determine the scope of the edges:

1) $\{R\} \longrightarrow \{T\}$

2) $\{Q\} \longrightarrow \{R\}$

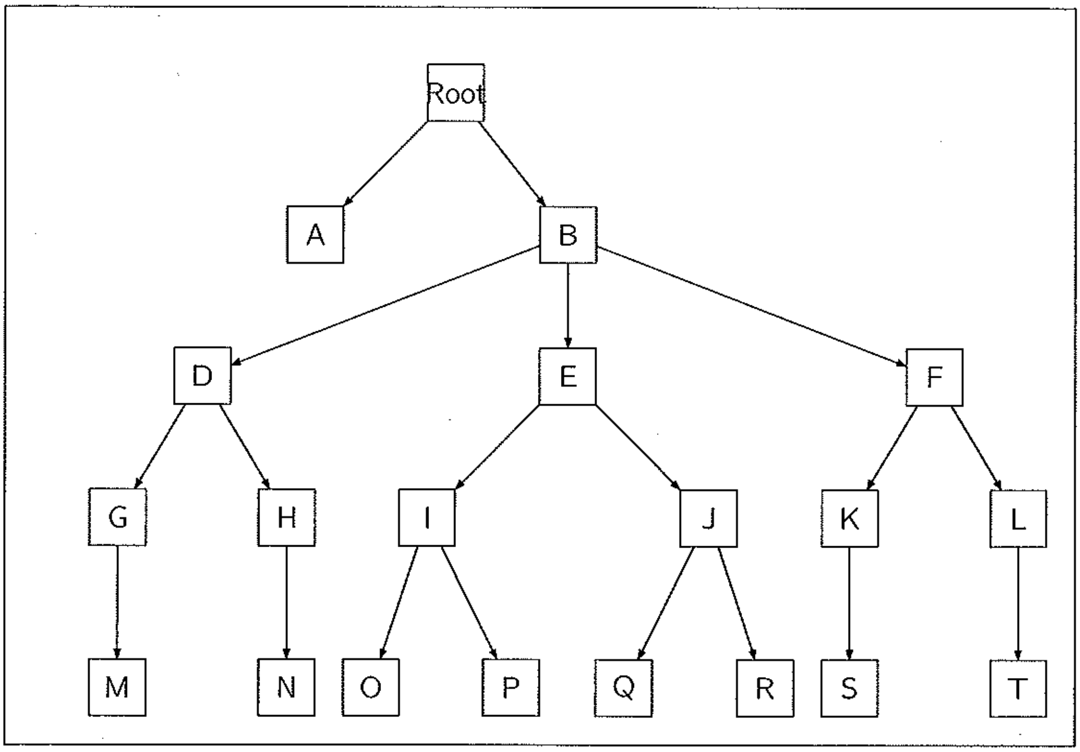
e) Consider the configuration $C = \{Root, B, E, O, Q\}$ in the statechart $\mathbb{S}$.

1) Provide the maximal set of events I that can be consumed in the configuration C.
2) Provide all possible steps in configuration C.

Solution:

a) $type(A) = BASIC$, $type(B) = OR$, $type(D) = AND$ and $type(N) = BASIC$.

b) The node hierarchy for $\mathbb{S}$ is as:



c) Priority of edges

- $scope(\{N\} \to \{Q\}) = B$, $scope(\{N\} \to \{Root\}) = Root$. Node B is an ancestor of node $Root$, therefore edge $\{N\} \to \{Q\}$ has a higher priority.
- $scope(\{M\} \to \{M\}) = G$, $scope(\{M\} \to \{O\}) = B$. Node G is an ancestor of node $Root$, therefore edge $\{M\} \to \{M\}$ has a higher priority.

d) Scope of an edge is the most nested OR-node that is unaffected by executing the edge.

1) $scope(\{R\} \longrightarrow \{T\}) = B$

2) $scope(\{Q\} \longrightarrow \{R\}) = J$

e) Consider the configuration $C = \{Root, B, E, O, Q\}$ in the statechart $\mathbb{S}$.

1) The maximal set of events $I = \{r, p\}$

2) The set of all possible steps is $Steps = \{O \overset{r/p}{\rightarrow} P, Q \overset{p/m}{\rightarrow} R\}$. This set is consistent as both the edges can be executed concurrently.