Exercise 12-1 Petri Nets

Explain if the following graphs are petri nets, workflow nets, or even sound workflow nets. Further express the graph as a sound workflow net if the graph is not yet a sound workflow net.

(a)  

This is not a Petri net, since between the transitions b and c, resp. b and d there is a missing place.

(b)  

This is a Petri net and a workflow net, but it is not sound, since there are deadlocks. The OR-Split after the transition a gets closed by an AND-JOIN which causes a deadlock. Transition d can not fire since it will always only get one token from eigther b or c.

(c)
This is a Petri net and a Workflow net but it is not sound, due to the violation of the proper completion constraint. It can terminate while there is still a token in an other place than the output place.

Exercise 12-2  $\alpha$-Miner

(a) For the $\alpha$-Miner algorithm, we use the relations $\succ$, $\rightarrow$, $\parallel$, $\#$ to denote direct successions, causality, parallelism or choice. Considering the set of activities \{a, b, c\}, give notion (graphically) about the following patterns and associate the right relations with them according to the activities having been used:

- **Sequence Pattern**
  Sequence: $a \rightarrow b$

- **XOR-Split and XOR-Join pattern**
  XOR-split: $a \rightarrow b, a \rightarrow c$ and $b \# c$
  XOR-join: $a \rightarrow c, b \rightarrow c$ and $b \# c$

- **AND-split and AND-join pattern**
  AND-split: $a \rightarrow b, a \rightarrow c$ and $b \parallel c$
  AND-join: $a \rightarrow c, b \rightarrow c$ and $a \parallel b$

(b) Given the trace $L_1 = \{(a, b, c, d), (a, c, b, d), (a, e, d)\}$. Determine the following sets:

- **Set of activities**: $T_L = \{t \in T \mid \exists \sigma \in L, t \in \sigma\}$
  Each activity in $L$ corresponds to a transition in $\alpha(L)$: $T_L = \{a, b, c, d, e\}$
• Set of start activities: \( T_I = \{ t \in T \mid \exists \sigma \in L t = first(\sigma) \} \)
  Fix the set of start activities - that is, the first elements of each trace: \( T_I = \{ a \} \)

• Set of end activities: \( T_O = \{ t \in T \mid \exists \sigma \in L t = last(\sigma) \} \)
  Fix the set of end activities - that is, elements that appear last at a trace: \( T_O = \{ d \} \)

• Set of paired activities:

\[
X_L = \{ (A, B) | A \subseteq T_L \land A \neq \emptyset \land
B \subseteq T_L \land B \neq \emptyset \land
\forall a \in A \forall b \in B a \rightarrow_L b \land
\forall a_1, a_2 \in A \#_1 a_2 \land \forall b_1, b_2 \in B b_1 \#_2 b_2 \}
\]

Find pairs \( (A, B) \) of sets of activities such that every element \( a \in A \) and every element \( b \in B \) are causally related (i.e. \( a \rightarrow_L b \)), all elements in \( A \) are independent (\( a_1 \#_1 a_2 \)), and all elements in \( B \) are independent (\( b_1 \#_2 b_2 \)) as well:

\[
X_L = \{ \{\{a\}, \{b\}\}, \{\{a\}, \{c\}\}, \{\{b\}, \{d\}\}, \{\{c\}, \{d\}\} \}
\]

• Set of paired activities that are maximal:

\[
Y_L = \{ (A, B) \in X_L \mid \forall (A', B') \in X_L A \subseteq A' \land B \subseteq B' \implies (A, B) = (A', B') \}
\]

Delete from \( X_L \) all pairs \( (A, B) \) that are not maximal.

\[
Y_L = \{ \{\{a\}, \{b\}\}, \{\{a\}, \{c\}\}, \{\{b\}, \{d\}\}, \{\{c\}, \{d\}\} \}
\]

• Set of places: \( P_L = \{ p(A, B) \mid (A, B) \in Y_L \} \cup \{ i_L, o_L \} \)

Determine the place set: Each element \( (A, B) \) of \( Y_L \) is a place. To ensure the workflow structure, add a source place \( i_L \) and a target place \( o_L \). \( P_L = \{ p(A, B), p(i_L, \{b,e\}), p(i_L, \{c,e\}), p(b, \{d\}), p(c, \{d\}) \} \)

• Flow relations:

\[
F_L = \{ (A, B, p(A, B)) \mid (A, B) \in Y_L \land a \in A \} \cup
\{ (p(A, B), b) \mid (A, B) \in Y_L \land b \in B \} \cup
\{ (i_L, t) \mid t \in T_I \} \cup \{ (t, o_L) \mid t \in T_O \}
\]

Determine the flow relation: Connect each place \( p(A, B) \) with each element \( a \) of its set \( A \) or source transition and with each element of its set \( B \) of target transitions. In addition, draw an arc from the source place \( i_L \) to each start transition \( t \in T_I \) and an arc from each end transition \( t \in T_O \) to the sink place \( o_L \).
• Definition (no task): \( \alpha \)-Miner on event log \( L \) is then defined as: \( \alpha(L) = (P_L, T_L, F_L) \)

(c) Construct the Footprint Table for trace \( L_1 \).

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>#L_1</td>
<td>( \rightarrow L_1 )</td>
<td>( \rightarrow L_1 )</td>
<td>#L_1</td>
<td>( \rightarrow L_1 )</td>
</tr>
<tr>
<td>b</td>
<td>( \leftarrow L_1 )</td>
<td>#L_1</td>
<td>( \parallel L_1 )</td>
<td>( \rightarrow L_1 )</td>
<td>#L_1</td>
</tr>
<tr>
<td>c</td>
<td>( \leftarrow L_1 )</td>
<td>( \parallel L_1 )</td>
<td>#L_1</td>
<td>( \rightarrow L_1 )</td>
<td>#L_1</td>
</tr>
<tr>
<td>d</td>
<td>#L_1</td>
<td>( \leftarrow L_1 )</td>
<td>#L_1</td>
<td>( \leftarrow L_1 )</td>
<td>#L_1</td>
</tr>
<tr>
<td>e</td>
<td>( \leftarrow L_1 )</td>
<td>#L_1</td>
<td>#L_1</td>
<td>( \rightarrow L_1 )</td>
<td>#L_1</td>
</tr>
</tbody>
</table>