MODULE *Example4_Choice*

Choice in a Petri Net. Either $t1$ fires or $t2$ fires. If $t1$ fires and we have a token at $p1$, that leads to having a token at $sink1$.

```
           --           --
         |t1| -> p1 -> |t3| -> sink1
      ->  --           --
source
      ->  --           --
         |t2| -> p2 -> |t4| -> sink2
           --           --
```

$Places \triangleq \{\text{“source”}, \text{“p1”}, \text{“p2”}, \text{“sink1”}, \text{“sink2”}\}$ Define the net.
$Transitions \triangleq \{\text{“t1”}, \text{“t2”}, \text{“t3”}, \text{“t4”}\}$
$Arcs \triangleq [$
$\quad source \mapsto \{\text{“t1”}, \text{“t2”}\},$
$\quad p1 \mapsto \{\text{“t3”}\},$
$\quad p2 \mapsto \{\text{“t4”}\},$

$\quad t1 \mapsto \{\text{“p1”}\},$
$\quad t2 \mapsto \{\text{“p2”}\},$
$\quad t3 \mapsto \{\text{“sink1”}\},$
$\quad t4 \mapsto \{\text{“sink2”}\}$
$]$
$ArcWeights \triangleq \langle\rangle$ Unspecified arc weights default to 1.
$InitialMarking \triangleq [source \mapsto 1]$
VARIABLE *Marking*

$PN \triangleq$ INSTANCE *PetriNet* Instantiate it within a namespace.

$Spec \triangleq PN!Spec$ Make *Spec* and *Invariants* available for the config file.

$Invariants \triangleq PN!Invariants$

Properties

Either eventually always $sink1$ or $sink2$.
$FinalMarking \triangleq PN!FinalMarking([sink1 \mapsto 1]) \lor PN!FinalMarking([sink2 \mapsto 1])$

Check that a marking leads to another.
$P1LeadToSink1 \triangleq Marking = PN!^*([p1 \mapsto 1]) \rightsquigarrow Marking = PN!^*([sink1 \mapsto 1])$

$IsStateMachine \triangleq PN!IsStateMachine$

$IsFreeChoiceNet \triangleq PN!IsFreeChoiceNet$