MODULE *RelationUtils*

Relation related operators.

LOCAL INSTANCE *Naturals*
LOCAL INSTANCE *FiniteSets*
LOCAL INSTANCE *Sequences*
LOCAL INSTANCE *SequencesExt*
LOCAL INSTANCE *Functions*

Basic definitions.

$Dom(R) \triangleq \{a : \langle a, b \rangle \in R\}$ Domain of R
$Ran(R) \triangleq \{b : \langle a, b \rangle \in R\}$ Range of R
$Support(R) \triangleq Dom(R) \cup Ran(R)$ Support of R

Basic operations.

$Image(R, a) \triangleq \{b \in Ran(R) : \langle a, b \rangle \in R\}$
$LeftRestriction(R, a) \triangleq \{\langle a, b \rangle : b \in Image(R, a)\}$

$InverseRelation(R) \triangleq \{\langle b, a \rangle : \langle a, b \rangle \in R\}$
$InverseImage(R, b) \triangleq \{a \in Dom(R) : \langle a, b \rangle \in R\}$

$R \mid S \triangleq R \cap (S \times S)$ Restriction of R on S

$R ** T \triangleq$ Composition of R and T
  LET $SR \triangleq Support(R)$
    $ST \triangleq Support(T)$
  IN $\{\langle r, t \rangle \in SR \times ST : \exists s \in SR \cap ST : (\langle r, s \rangle \in R) \land (\langle s, t \rangle \in T)\}$

$GT(R, a) \triangleq \{b \in Ran(R) : \langle a, b \rangle \in R\}$ $\triangleq Image(R, a)$
$LT(R, b) \triangleq \{a \in Dom(R) : \langle a, b \rangle \in R\}$ $\triangleq InverseImage(R, b)$

The following definition is from https://*github.com*/jameshfisher/tlaplus/blob/master/examples/*TransitiveClosure*/*TransitiveClos*
It also contains several other methods for computing TC.

$TC(R) \triangleq$ Transitive closure of R
  LET $S \triangleq Support(R)$
    RECURSIVE $TCR(\_)$
    $TCR(T) \triangleq$ IF $T = \{\}$
        THEN R
        ELSE LET $r \triangleq$ CHOOSE $s \in T$: TRUE
            $RR \triangleq TCR(T \setminus \{r\})$
          IN $RR \cup \{\langle s, t \rangle \in S \times S :$
              $\langle s, r \rangle \in RR \land \langle r, t \rangle \in RR\}$
  IN $TCR(S)$

Example: $SeqToRel(\langle 1, 2, 3 \rangle) = \{\langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 2, 3 \rangle\}$

RECURSIVE $Seq2Rel(\_)$
$Seq2Rel(s) \triangleq$ Transform a sequence s into a strict total order relation
  IF $s = \langle \rangle$ THEN $\{\}$

```
        ELSE LET h ≜ Head(s)
                 t ≜ Tail(s)
             IN  {⟨h, r⟩ : r ∈ Range(t)} ∪ Seq2Rel(t)
────────────────────────────────────────────────────────────────
Basic properties.
IsReflexive(R, S) ≜ ∀ a ∈ S : ⟨a, a⟩ ∈ R
IsIrreflexive(R, S) ≜ ∀ a ∈ S : ⟨a, a⟩ ∉ R

IsSymmetric(R, S) ≜ ∀ a, b ∈ S : ⟨a, b⟩ ∈ R ≡ ⟨b, a⟩ ∈ R
IsAntisymmetric(R, S) ≜ ∀ a, b ∈ S : ⟨a, b⟩ ∈ R ∧ ⟨b, a⟩ ∈ R ⇒ a = b

IsTransitive(R, S) ≜
    ∀ a, b, c ∈ S : (⟨a, b⟩ ∈ R ∧ ⟨b, c⟩ ∈ R) ⇒ ⟨a, c⟩ ∈ R

IsTotal(R, S) ≜
    ∀ a, b ∈ S : ⟨a, b⟩ ∈ R ∨ ⟨b, a⟩ ∈ R

IsPartialOrder(R, S) ≜
    ∧ IsReflexive(R, S)
    ∧ IsAntisymmetric(R, S)
    ∧ IsTransitive(R, S)

IsTotalOrder(R, S) ≜
    ∧ IsPartialOrder(R, S)
    ∧ IsTotal(R, S)

IsStrictPartialOrder(R, S) ≜
    ∧ IsIrreflexive(R, S)
    ∧ IsTransitive(R, S)

IsStrictTotalOrder(R, S) ≜
    ∧ IsStrictPartialOrder(R, S)
    ∧ IsTotal(R, S)

Respect(R, T) ≜ T ⊆ R    Does R respect T?
────────────────────────────────────────────────────────────────
Special elements in a relation
Minimal(R, S) ≜    the set of minimal elements in relation R on the set S
    {m ∈ S : ¬∃ a ∈ Dom(R) : ⟨a, m⟩ ∈ R}
Maximal(R, S) ≜    the set of maximal elements in relation R on the set S
    {m ∈ S : ¬∃ b ∈ Ran(R) : ⟨m, b⟩ ∈ R}
────────────────────────────────────────────────────────────────
AnyLinearExtension(R, S) ≜    return an arbitrary linear extension of R on the set S
    LET RECURSIVE LinearExtensionUtil(_, _)
        LinearExtensionUtil(rel, set) ≜    rel: remaining relation; set: remaining set
            IF set = {} THEN ⟨⟩
             ELSE LET m ≜ CHOOSE x ∈ Minimal(rel, set) : TRUE
```

```
                    IN  ⟨m⟩ ∘ LinearExtensionUtil(rel \ LeftRestriction(R, m), set \ {m})
    IN  LinearExtensionUtil(R, S)

 See https://groups.google.com/g/tlaplus/c/mtyEmqhlRVg
AllLinearExtensions(R, S) ≜   return all possible linear extensions of R on the set S
    LET RECURSIVE LinearExtensionsUtil(_, _)
        LinearExtensionsUtil(rel, set) ≜
            IF set = {} THEN {⟨⟩}
             ELSE LET Extend(m) ≜ {⟨m⟩ ∘ l :   extend recursively by the minimal element m
                            l ∈ LinearExtensionsUtil(rel \ LeftRestriction(R, m), set \ {m})}
                  IN  UNION {Extend(m) : m ∈ Minimal(rel, set)}   for each minimal element
    IN  LinearExtensionsUtil(R, S)

LinearExtensions(R, S) ≜   return the set of all possible linear extensions of R on the set S
    {l ∈ TupleOf(S, Cardinality(S)) : Respect(Seq2Rel(l), R)}
├──────────────────────────────────────────────────────────────────────────────┤
 Test cases
set1 ≜ {2, 3, 5, 7, 8, 9, 10, 11}
rel1 ≜   from https://en.wikipedia.org/wiki/Topological_sorting
    {⟨3, 8⟩, ⟨3, 10⟩, ⟨5, 11⟩, ⟨7, 8⟩, ⟨7, 11⟩,
     ⟨8, 9⟩, ⟨11, 2⟩, ⟨11, 9⟩, ⟨11, 10⟩}

set2 ≜ 0 . . 5
rel2 ≜   from https://www.geeksforgeeks.org/topological-sorting/
    {⟨2, 3⟩, ⟨3, 1⟩, ⟨4, 0⟩, ⟨4, 1⟩, ⟨5, 0⟩, ⟨5, 2⟩}

set3 ≜ 1 . . 6
rel3 ≜   from https://leetcode.com/discuss/general-discussion/1078072/introduction-to-topological-sort
    {⟨1, 2⟩, ⟨1, 4⟩, ⟨2, 3⟩, ⟨4, 2⟩, ⟨4, 5⟩, ⟨4, 6⟩, ⟨5, 6⟩}
├──────────────────────────────────────────────────────────────────────────────┤
THEOREM LE ≜
     ∧ AllLinearExtensions(rel1, set1) = LinearExtensions(rel1, set1)
     ∧ AllLinearExtensions(rel2, set2) = LinearExtensions(rel2, set2)
     ∧ AllLinearExtensions(rel3, set3) = LinearExtensions(rel3, set3)
└──────────────────────────────────────────────────────────────────────────────┘
\* Modification History
\* Last modified Sun Apr 18 10:50:36 CST 2021 by hengxin
\* Created Tue Sep 18 19:16:04 CST 2018 by hengxin
```