```
MODULE OT -
 1
    Specification of OT (Operational Transformation) functions. It consists of the basic OT functions
    for two operations and more general ones involving operation sequences.
   EXTENDS Op Operators, Set Utils
    OT (Operational Transformation) functions.
    Naming convention: I for "Ins" and D for "Del".
    The left "Ins" lins transformed against the right "Ins" rins.
    X form II(lins, rins) \stackrel{\triangle}{=}
18
        IF lins.pos < rins.pos
19
         THEN lins
20
         ELSE IF lins.pos > rins.pos
21
                 THEN [lins EXCEPT !.pos = @ + 1]
22
                 ELSE IF lins.ch = rins.ch
23
                          THEN Nop
24
                          ELSE IF lins.pr > rins.pr
25
                                  THEN [lins EXCEPT !.pos = @ + 1]
26
                                  ELSE lins
27
    The left "Ins" ins transformed against the right "Del" del.
    X form ID(ins, del) \triangleq
31
32
        IF ins.pos \leq del.pos
         Then ins
33
         ELSE [ins EXCEPT !.pos = @-1]
34
    The left "Del" del transformed against the right "Ins" ins.
    X form DI(del, ins) \triangleq
38
        If del.pos < ins.pos
39
         THEN del
40
         ELSE [del \ EXCEPT \ !.pos = @ + 1]
41
    The left "Del" ldel transformed against the right "Del" rdel
    XformDD(ldel, rdel) \stackrel{\triangle}{=}
45
        If ldel.pos < rdel.pos
46
47
         THEN ldel
         ELSE IF ldel.pos > rdel.pos
48
                 THEN [ldel EXCEPT !.pos = @ -1]
49
                 ELSE Nop
50
51
    Transform the left operation lop against the right operation rop with appropriate OT function.
    Xform(lop, rop) \triangleq
56
        Case lop = Nop \lor rop = Nop \to lop
57
58
           \square lop.type = "Ins" \land rop.type = "Ins" \rightarrow XformII(lop, rop)
           \Box \ lop.type = "Ins" \ \land \ rop.type = "Del" \ \rightarrow X form ID(lop, \ rop)
59
           \square lop.type = "Del" \land rop.type = "Ins" \rightarrow XformDI(lop, rop)
60
```

```
\square lop.type = "Del" \land rop.type = "Del" \rightarrow XformDD(lop, rop)
 61
 62 F
     Generalized OT functions on operation sequences
     Iteratively/recursively transforms the operation op against an operation sequence ops.
    RECURSIVE XformOpOps(\_, \_)
     X form Op Ops(op, ops) \stackrel{\Delta}{=}
         IF ops = \langle \rangle
 73
 74
               THEN op
               ELSE X form Op Ops(X form(op, Head(ops)), Tail(ops))
 75
     Iteratively/recursively transforms the operation op against an operation sequence ops. Being
     different from XformOpOps, XformOpOpSX maintains the intermediate transformed operation
     RECURSIVE XformOpOpsX(\_, \_)
     X form Op Ops X(op, ops) \stackrel{\Delta}{=}
         IF ops = \langle \rangle
 84
 85
               THEN \langle op \rangle
               ELSE \langle op \rangle \circ XformOpOpsX(Xform(op, Head(ops)), Tail(ops))
 86
     Iteratively/recursively transforms the operation sequence ops against an operation op.
     XformOpsOp(ops, op) \triangleq
 91
         LET opX \stackrel{\triangle}{=} XformOpOpsX(op, ops)
 92
               [i \in 1 .. Len(ops) \mapsto Xform(ops[i], opX[i])]
 93
     Iteratively/recursively transforms an operation sequence ops1 against another operation sequence
     ops2.
     See also Definition 2.13 of the paper "Imine @ TCS06".
    RECURSIVE XformOpsOps(\_, \_)
100
     X form Ops Ops (ops 1, ops 2) \stackrel{\Delta}{=}
101
         IF ops2 = \langle \rangle
102
103
          THEN ops1
          ELSE X form Ops Ops(X form Ops Op(ops1, Head(ops2)), Tail(ops2))
104
105
     \* Modification History
     \* Last modified Fri Dec 28 10:37:02 CST 2018 by hengxin
     \* Created Sun Jun 24 15:57:48 CST 2018 by hengxin
```