```
MODULE Op
```

Model checking basic operations on strings (*i.e.*, list of characters).

```
EXTENDS Naturals, Sequences

CONSTANTS Char

List ≜ Seq(Char)    The set of all lists.
```

The set of all operations. In this specification, we will focus on "Ins" and "Del".

```
Op ≜ [type : {"Rd"}] ∪                a read specifies no arguments
     [type : {"Del"}, pos : Nat \ {0}] ∪    a deletion specifies a position (from 1)
     [type : {"Ins"}, pos : Nat \ {0}, ch : Char, pr : Nat]    an insertion specifies a position (from 1), a character, and
Nop ≜ CHOOSE v : v ∉ Op    Nop: an operation representing "doing nothing"
```

Some operations for test.

```
Del1 ≜ [type ↦ "Del", pos ↦ 1]
Del2 ≜ [type ↦ "Del", pos ↦ 2]
Del3 ≜ [type ↦ "Del", pos ↦ 3]
Ins1 ≜ [type ↦ "Ins", pos ↦ 1, ch ↦ "a", pr ↦ 1]
Ins2 ≜ [type ↦ "Ins", pos ↦ 2, ch ↦ "b", pr ↦ 2]
Ins3 ≜ [type ↦ "Ins", pos ↦ 3, ch ↦ "c", pr ↦ 3]
Ops ≜ ⟨Ins2, Del3, Ins1, Del2, Ins3, Del1⟩
```

The "Apply" operator which applies an operation *op* on the list *l*.

```
Apply(op, l) ≜
    LET len ≜ Len(l)
        pos ≜ op.pos
    IN  CASE op.type = "Del" → SubSeq(l, 1, pos − 1) ∘ SubSeq(l, pos + 1, len)
          □  op.type = "Ins" → Append(SubSeq(l, 1, pos − 1), op.ch) ∘ SubSeq(l, pos, len)
```

* Modification History
* Last modified Sun *Jun* 24 17:16:55 *CST* 2018 by *hengxin*
* Created Sat *Jun* 23 20:56:53 *CST* 2018 by *hengxin*