

# Arrays, Lists, and Sequences



# Arrays, Lists, and Sequences



- ⦿ To represent a conventional array or a list, we simply use a table with integer keys.
- ⦿ There is neither a way nor a need to declare a size, we just initialize the elements we need.

# Arrays, Lists, and Sequences



- ⊙ Usually, when we manipulate a list we should know its length.
- ⊙ For instance, after we read 5 lines into a list, it is easy to know that its length is 5, because its numeric keys are 1,2,...5
- ⊙ This technique only works if the list does not have *holes* , which are nil elements inside it. We call such a list without holes a *sequence*.

# Sequences



- ⊙ For sequences, Lua offers the length operator (#).
- ⊙ On tables, it gives the length of the *sequence* represented by the table.
- ⊙ Formally, a *sequence* is a table where the positive numeric keys comprise a set  $\{1, \dots, n\}$  for some  $n$ .

# Sequences



- ⦿ The length operator also provides a useful idiom for manipulating sequences:

```
a[#a + 1] = p -- appends 'p' to the end of the sequence
```

- ⦿ It is important to bear in mind that the length operator is unreliable for lists with holes (nils). It only works on sequences.