ARRAYED_CONTAINER

```
feature -- Constructors
  make
      -- Initialize an empty container.
    ensure
      empty_container: imp.is_empty

feature -- Commands
  assign_at (i: INTEGER; s: STRING)
     -- Change the value at position 'i' to 's'.
    require
      valid_index: valid_index (i)
    ensure
      size_unchanged: imp.count = (old imp.twin).count
      item_assigned: imp [i] ~ s
      others_unchanged: ∀j : 1 ≤ j ≤ imp.count : j ≠ i ⇒ imp[j] ~ (old imp.twin)[j]

  delete_at (i: INTEGER)
      -- Delete element stored at index 'i'.
    require
      valid_index: valid_index (i)
    ensure
      size_changed: imp.count = (old imp.twin).count - 1
      left_half_the_same: ∀j : 1 ≤ j ≤ (i - 1) :  imp[j] ~ (old imp.twin)[j]
      right_half_the_same: ∀j : i ≤ j ≤  imp.count :  imp[j] ~ (old imp.twin)[(j + 1)]

  insert_at (i: INTEGER; s: STRING)
      -- Insert value 's' into index 'i'.
   require
      valid_index: valid_index (i)
   ensure
      size_changed: imp.count = (old imp.twin).count + 1
      inserted_at_i: imp [i] ~ s
      left_half_the_same: ∀j : 1 ≤ j ≤ (i - 1) :  imp[j] ~ (old imp.twin)[j]
      right_half_the_same: ∀j : (i + 1) ≤ j ≤  imp.count :  imp[j] ~ (old imp.twin)[(j - 1)]

  insert_last (s: STRING)
      -- Insert 's' as the last element of the container.
   ensure
      size_changed: imp.count = (old imp.twin).count + 1
      last_inserted: imp [imp.count] ~ s
      others_unchanged: ∀j : 1 ≤ j ≤  imp.count :  imp[j] ~ (old imp.twin)[j]

  remove_first
      -- Remove first element from the container.
   require
      not_empty: (count > 0)
   ensure
      size_changed: imp.count = (old imp.twin).count - 1
      others_unchanged: ∀j : 1 ≤ j ≤  imp.count :  imp[j] ~ (old imp.twin)[(j + 1)]

feature -- Queries
  count: INTEGER
      -- Number of items currently stored in the container.

  get_at (i: INTEGER): STRING
      -- Return the element stored at index 'i'.
   require
      valid_index: valid_index (i)
   ensure
      size_unchanged: imp.count ~ (old imp.twin).count
      result_correct: Result ~ (old imp.twin) [i]
      no_elements_changed: ∀j : 1 ≤ j ≤  imp.count :  imp[j] ~ (old imp.twin)[j]

  valid_index (i: INTEGER): BOOLEAN
      -- Is 'i' a valid index of current container?
   ensure
      size_unchanged: imp.count ~ (old imp.twin).count
      result_correct: Result ~ ((old imp.twin).lower <= i and i <= (old imp.twin).upper)
      no_elements_changed: ∀j : 1 ≤ j ≤  imp.count :  imp[j] ~ (old imp.twin)[j]

feature -- {NONE}
  -- Implementation of container via an array
  imp : ARRAY[STRING]

invariant
  consistency: imp.count = count
```