2.4 Polynomials

2.4.1 The Abstract Data Type

- We can use arrays to implement other abstract data-types.
- Let us consider the [ordered or linear list] data structure.

Examples

- Days of the week: (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday)

- Values in a deck of cards: (Ace, 2, 3, 4, ..., 9, 10, Jack, Queen, King)

- Floors of a building: (basement, lobby, mezzanine, first, second)

- Years the United States fought in World War II: (1941, 1942, 1943, 1944, 1945)

- Years Switzerland fought in World War II: () → empty list

General form: $(item_0, item_1, ..., item_{n-1})$

Operations on lists

1) Finding the length, n, of a list 2) Reading the items in a list from left to right (or right to left). 3) Retrieving the ith item from a list, $0 \leq i < n$.

4) Replacing the item in the ith position of a list, $0 \leq i < n$.

5) Inserting a new item in the ith position of a list, $0 \leq i < n$. The items previously numbered $i, i+1, ..., n-1$ become items numbered $i+1, i+2, ..., n$.

6) Deleting an item from the ith position of a list, $0 \leq i < n$. The items numbered $i+1, ..., n-1$ become items numbered $i, i+1, ..., n-2$.

Implementation

- We represent an ordered list as an array where we associate the list element, $item_i$, with the array index i. Assuming the standard implementation of array, the mapping is sequential.