

A First Book of C++

From Here to There

Arrays

One-Dimension Arrays

- One-Dimension Array(Single-Dimension Array or Vector): a list of related values
 - All items in list have same data type
 - All list members stored using single group name
- Example: a list of grades
98, 87, 92, 79, 85
 - All grades are integers and must be declared
 - Can be declared as single unit under a common name (the array name)

One-Dimension Arrays (continued)

- Array declaration statement provides:
 - The array(list) name
 - The data type of array items
 - The number of items in array
- Syntax
`dataType arrayName [numberOfItems]`
 - Common programming practice requires defining number of array items as a constant before declaring the array

One-Dimension Arrays (continued)

- Examples of array declaration statements:

```
const int NUMELS = 5; // define a constant
                    // for the number of
                    // items
int grade[NUMELS]; // declare the array

const int ARRAYSIZE = 4;
char code[ARRAYSIZE];

const int NUMELS = 6;
double prices[NUMELS];
```

A First Book of C++: From Hello To There, Third Edition

4

One-Dimension Arrays (continued)

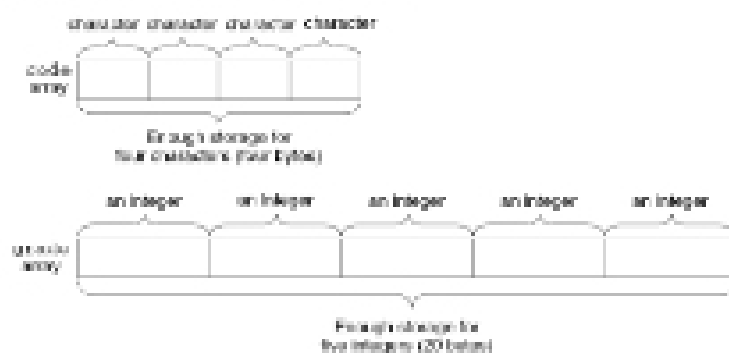
- Each array allocates sufficient memory to hold the number of data items given in declaration
- Array element(component): an item of the array
- Individual array elements stored sequentially
 - A key feature of arrays that provides a simple mechanism for easily locating single elements

A First Book of C++: From Hello To There, Third Edition

5

One-Dimension Arrays (continued)

FIGURE 5.3 The grade and code arrays in memory



A First Book of C++: From Hello To There, Third Edition

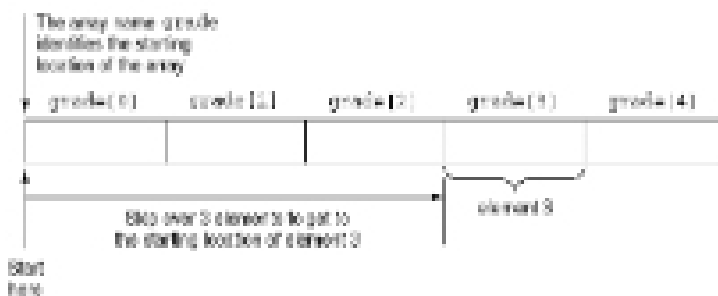
6

One-Dimension Arrays (continued)

- **Index (subscript value):** position of individual element in an array
- **Accessing of array elements:** done by giving array name and element's index
 - `grade[0]` refers to first grade stored in `grade` array
- **Subscripted variables can be used anywhere that scalar variables are valid:**
 - `grade[0] = 95.75;`
 - `grade[1] = grade[0] - 11.0;`

One-Dimension Arrays (continued)

FIGURE 85 Accessing an Individual Array Element—Element 3



One-Dimension Arrays (continued)

- **Subscripts:** do not have to be integers
 - Any expression that evaluates to an integer may be used as a subscript
 - Subscript must be within the declared range
- **Examples of valid subscripted variables** (assumes `i` and `j` are `int` variables):
 - `grade[i]`
 - `grade[2*i]`
 - `grade[j-1]`
