

CS221: Computer Programming I

B.Sc in Computer Science

Autumn 2011

CS221: Computer Programming I

Chapter IV: Arrays, Pointers and Strings

In this chapter ...

- Array
- Pointer
- String

Array

- What it is about
- Array declaration & initialization
- Accessing array elements
- Multidimensional array

Array

- Structured data type
- Represents an indexed list
- An indexed list is a list of ordered elements randomly accessed through their position in the list (ie. index).
- An array groups a number of elements that are accessed thru a common variable identifier – which is the array name.

Array declaration

Syntax

```
<type>      <idf> [ <size> ];
```

Array declaration

Description

- `<size>` is an integer constant expression representing the maximum number of elements to be accommodated.
- `<idf>` is the common array name.
- The elements of the array are accessed as follows: `<idf>[0]`, `<idf>[1]`, ... , `<idf>[<size> - 1]`
- Total memory space reserved = `<size> * sizeof(<type>)`

Array declaration: Example

```
short x[10];  
float p[45];  
double velocity[1200];
```

Array declaration: Example

- `short x[10];`
- → max number of elements = 10
- → x is the common array name
- → `x[0]`, `x[1]`, `x[2]`, ..., `x[9]` are the elements of the array x, each are of short type.
- → Memory reserved = $10 * 2 = 20$ bytes

Array declaration: Example

- `short x[10];`
- `→ x[0] = 6;`
- `→ x[5] = 23478;`
- `→ x[2] = x[0] + 12;`
- `→ cin >> x[1];`
- `→ cout << x[8]*x[3];`
- `→ x[10], x[11], x[-1] are all illegal indices`

Array Example

- `Void main()`
- `{ float x[35], sum, average;`
- `int n;`
- `cout << "Enter the size of the array > ";`
- `cin >> n;`
- `sum = 0;`
- `for (int i=0; i<=n-1; i++) {`
 - `cout << "Enter data element " << i + 1 << " ";`
 - `cin >> x[i];`
 - `sum = sum + x[i];`
- `}`
- `average = sum/n;`
- `cout << "The average is = " << average << endl;`
- `}`

Questions on the previous example

- What will happen if the size of the array as entered from the user (ie. n) exceeds the maximum reserved space (ie. 35)?
- What solution could you propose for Q1?
- What will happen if the size of the array as entered from the user (ie. n) is zero?

Exercise

Modify the above example so that it counts for the number of elements below the average.

Is this solution correct?

- `Void main()`
- `{ float x, sum, average;`
- `int n;`
- `cout << "Enter the size of the array > ";`
- `cin >> n;`
- `sum = 0;`
- `for (int i=0; i<=n-1; i++) {`
 - `cout << "Enter data element " << i + 1 << " ";`
 - `cin >> x;`
 - `sum = sum + x;`
- `}`
- `average = sum/n;`
- `count = 0;`
- `for (i = 0; i <= n-1; i++)`
 - `if (x < average) count++;`
- `cout << "The number of elements below the average = " << count`
`<< endl;`
- `}`

Correct Solution

- `void main()`
- `{ float x[35], sum, average;`
- `int n;`
- `cout << "Enter the size of the array > ";`
- `cin >> n;`
- `sum = 0;`
- `for (int i=0; i<=n-1; i++) {`
 - `cout << "Enter data element " << i + 1 << " ";`
 - `cin >> x[i];`
 - `sum = sum + x[i];`
- `}`
- `average = sum/n;`
- `count = 0;`
- `for (i = 0; i <= n-1; i++)`
 - `if (x[i] < average) count++;`
- `cout << "The number of elements below the average = " << count`
`<< endl;`
- `}`

Exercise

Write a program that cyclically shifts the elements of an array one position to the left (or to the right).