Programming Fundamentals
Lecture #11
Arrays

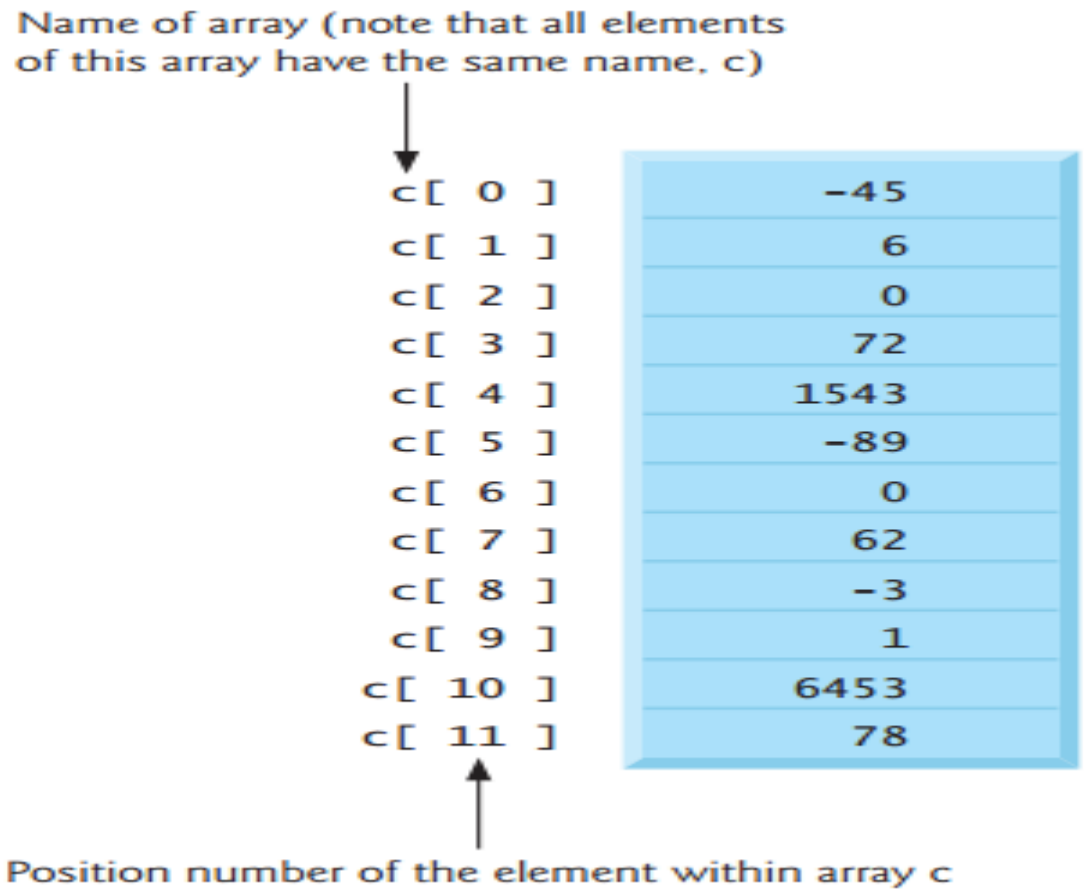
Junaid Hassan

Lecturer CS & IT Department UOS MBDIN

junaidte14@gmail.com

Introduction to Arrays:

- Arrays are data structures consisting of related data items of the same type
- An array is a group of memory locations related by the fact that they all have the same name and the same type
- A type of variable which can hold multiple values



- This array contains 12 elements
- Any one of these elements may be referred to by giving the name of the array followed by the position number of the particular element in square brackets ([])

- The first element in every array is the zeroth element. Thus, the first element of array c is referred to as c[0], the second element of array c is referred to as c[1], the seventh element of array c is referred to as c[6], and, in general, the ith element of array c is referred to as c[i – 1]
- Array names, like other variable names, can contain only letters, digits and underscores. Array names cannot begin with a digit

- The position number contained within square brackets is more formally called a subscript (or index). A subscript must be an integer or an integer expression.
- If a program uses an expression as a subscript, then the expression is evaluated to determine the subscript

For example, if a = 5 and b = 6, then the statement:

c[a + b] += 2;

adds 2 to array element c[11]

Defining Arrays

- Arrays occupy space in memory. You specify the type of each element and the number of elements required by each array so that the computer may reserve the appropriate amount of memory
- To tell the computer to reserve 12 elements for integer array c, the following definition is used:
- int c[12];

- The following definition
- int b[100], x[27];
- reserves 100 elements for integer array b and 27 elements for integer array x

```
int n[ 10 ]; /* n is an array of 10 integers */
int i; /* counter */

/* initialize elements of array n to 0 */
for ( i = 0; i < 10; i++ ) {
   n[ i ] = 0; /* set element at location i to 0 */
} /* end for */

printf( "%s%13s\n", "Element", "Value" );

/* output contents of array n in tabular format */
for ( i = 0; i < 10; i++ ) {
   printf( "%7d%13d\n", i, n[ i ] );
} /* end for */
```

- The other initializing methods:
- /* use initializer list to initialize array n */
  int n[10] = { 32, 27, 64, 18, 95, 14, 90, 70, 60, 37 };

- int n[10] = { 0 };
- int n[] = { 1, 2, 3, 4, 5 }; //number of array elements depends on the number of elements in initializer list
- int n[5] = { 32, 27, 64, 18, 95, 14 }; ——> Syntax error. why?

```
#define SIZE 10

/* symbolic constant SIZE can be used to specify array size */

int s[ SIZE ]; /* array s has SIZE elements */
```

Assignment #03

- Write down at least 20 Common Programming Errors mentioned in each chapter of the course book
- Assignment must be submitted in hand written form