Arrays and Strings

Introduction

- A fixed size sequenced collection of elements of the same data type
  - Simply a grouping of like-type data
  - E.g. list of employees in an organization
  - Marks of 100 students
  - List of customers and their telephone numbers
  - Table of daily rainfall data
- Arrays are structured data types
  - As they can be used to represent data values that have a structure of some sort

Introduction…

- Data Types
  - Derived data types
    - Arrays
    - Functions
    - Pointers
  - Fundamental data types
    - Integral types
    - Float types
    - Character types
  - User-defined data types
    - Structures
    - Unions
    - Enumerated data types

One-dimensional Arrays

- One variable with only one subscript
  - x[0],x[1],x[2],…,x[n]
  - 0,1,2,…n are subscripts and x is a variable
- Declaration

```
type variable-name[size];
float height[50];
char name[10];
```

- Size should be either numeric constant or a symbolic constant
- Any reference to the arrays outside the declared limits would not cause an error but might result in unpredictable program results…

One-dimensional Arrays…

```
char name[10];
```

- Declares the name as a character array variable that can hold a maximum of 10 characters.
- The last character in character array must be a NULL or ‘\0’ character.

‘W’
‘E’
‘L’
‘L’
‘ ‘
‘C’
‘O’
‘M’
‘E’
‘\0’

A program to illustrate a One-dimensional array

- A program that reads N values and computes the sum of their squares.
- A program that finds the maximum marks of students in a class of 100.

Initialization of one-d array

- Compile time
- Run time

Initialization of one-d array…

- Compile time initialization

```
type array-name[size] = {list of values};
int number[3]={0,0,0};
```

- Declare number as an array of size 3 and initialize each element to 0

```
float total[5]={0.0,15.3,6.2};
```

- Declare total as an array of size 5 and initialize first three elements to 0.0, 15.3 and 6.2 and remaining elements to 0.0

```
int number[]={0,0,0};
```

- Declare number as an array of size 3 and initialize each element to 0

Initialization of one-d array…

```
char
  name[6]={'h','e','l','l','o','\0'};
```

- Declare name as a character array of size 6 and initialize it to “hello”.
- Same as,

```
char name[6]="hello";
```

- The following is illegal in C

```
int x[2]={1,2,3,4};
```

More elements than array size !!!

Initialization of one-d array…

- Run time initialization

```
int x[10];
for(i=0;i<10;i++)
{
    printf(“Enter a number\n”);
    scanf(“%d”,&x[i]);
}
```

Two dimensional arrays

- When we need to store a table of values
- Consider the data structure that shows the value of sales of three items by four sales girls

	Item 1	Item 2	Item 3
Sales girl 1	310	234	110
Sales girl 2	112	345	321
Sales girl 3	102	321	213
Sales girl 4	250	321	321

- C allows us to define such table by using two dimensional array

Two dimensional arrays…

- Declaration

  type array-name[row_size][col_size];

- Representation of 2D array in memory

	Col 0	Col 1	Col 2
	[0][0]	**[0][1]**	**[0][2]**
Row 0	**310**	**275**	**365**
	[1][0]	[1][1]	[1][2]
Row 1	**10**	**190**	**325**
	[2][0]	[2][1]	[2][2]
Row 2	**405**	**235**	**240**
	[3][0]	[3][1]	[3][2]
Row 3	**310**	**275**	**365**

Initializing 2D array

- Run-time initialization

```
int sales[4][3];
for(int i=0;i<4;i++)
        for(int j=0;j<3;j++)
                scanf("%d", &sales[i][j]);
```

Initializing 2D array…

- Compile-time initialization
- int table[2][3] = {0,0,0,1,1,1}; initializes the elements of first row to 0 and second row to 1.
- int table[2][3]={{0,0,0},{1,1,1}}; does the same thing
- int table[][3]={ {0,0,0},{1,1,1}}; is permitted if all elements of the array are initialized. The statement will initialize table with 2 rows and 3 columns
- What about
  - int table[2][3]={{1,1},{0}};
  - int table[3][5]={{0},{0},{0}};
  - int table[3][5]={0};

A program illustrating 2D arrays

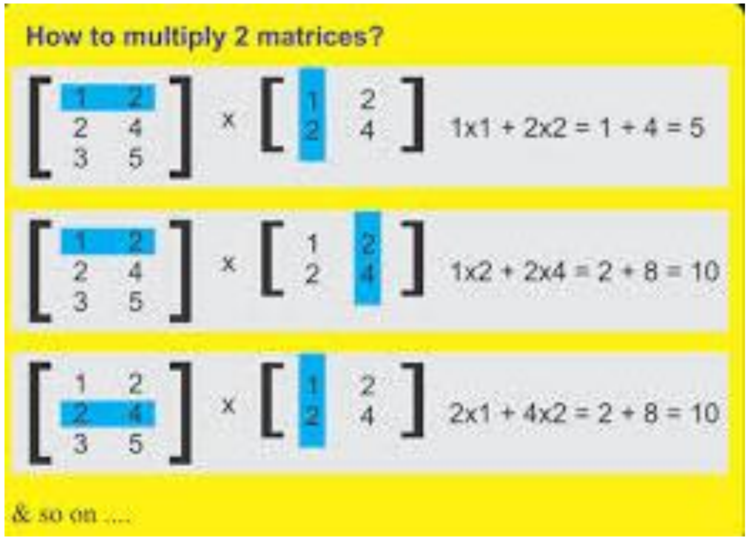
- Consider the data structure that shows the value of sales of three items by four sales girls

	Item 1	Item 2	Item 3
Sales girl 1	310	234	110
Sales girl 2	112	345	321
Sales girl 3	102	321	213
Sales girl 4	250	321	321

- Write a program that computes the following:
  - Total value of sales by each girl
  - Total value of each item sold
  - Grand total of sales of all items by all girls

A program illustrating 2D arrays…

- A program to multiply the elements of two NxN matrices



$$\begin{bmatrix}1&2\\2&4\\3&5\end{bmatrix} \times \begin{bmatrix}1&2\\2&4\end{bmatrix}$$

A program illustrating 2D arrays…

- A program to find the transpose of a matrix

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \quad \text{Original matrix}$$

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}^T \Rightarrow \begin{bmatrix} a & d & g \\ b & e & h \\ c & f & i \end{bmatrix}$$

Multi-dimensional array

- The general form

  type array_name[s1][s2][s3]…[sm];

  e.g. int survey[3][5][2];
  float table[5][4][5][3];

Dynamic arrays

- So far, we discussed static arrays…
  - Works fine as long as we know the size of the array
- Dynamic arrays allow us to specify the array size at run time..
  - Can be created using pointer variables and memory management functions such as malloc, calloc and realloc….

Character Arrays and Strings

Introduction to String

- String
  - A sequence of characters that is treated as a single data item
  - Have you used strings so far ???
  - What about printf(“Hello”); statement?
  - “Hello” is a string !!!

Introduction to String…

- Strings are often used to build meaningful and readable programs
- The common operations performed on character strings include:
  - Reading and writing strings
  - Combining strings together
  - Copying one string to another
  - Comparing strings for equality
  - Extracting a portion of a string

Declaring and initializing string variables

char string_name[size];

- The size determines the number of characters in a the string_name
- When compiler assigns a character string to a character array, it automatically supplies a NULL character ('\0') at the end of a string
- Therefore, the size should be the length of a string plus one

Declaring and initializing string variables

char city[9]= “Surat”;

char city[9]={‘S’,’U’,’R’,’A’,’T’,’\0’};

char name[]={‘g’,’o’,’o’,’d’,’\0’};

char name[10]=“good”; the size of name will be 10 only !!!

char str[5];

str=“good”;

illegal

char str1[5]=“good”;

char str2[5];

str2=str1;

illegal

Reading strings from terminal

- Using scanf function

```
char addr[20];
scanf("%s",addr);
```

- scanf terminates as soon as the first white space is found !!!
- What happens if you want to input “NEW YORK” in addr ???
- Only “NEW” will be stored in addr !!!

Reading strings from terminal…

- Using getchar and gets functions

```
char ch;
ch=getchar();
```

- A program that reads a line of text containing a series of words from the terminal

Reading strings from terminal…

- Using gets function

```
char str[20];
gets(str);
printf(“%s”,str);
```

- C does not provide operators that work on strings directly…
  - E.g. str = “ABC”;

    str1=str2;   are invalid

Programs illustrating strings

- A program to find length of a string
- A program to copy one string into another and count the number of characters copied
- A program that checks whether the string is palindrome or not.

String handling library functions

- strcat()
- strcmp()
- strcpy()
- strlen()

Assignment 4

- Chapter 6: Review Questions

6.1,6.2,6.8,6.9,6.11, 6.12,6.16 to 6.20