SIES: BASIC C PROGRAMMING

L#13: STRUCTURES

Seung Beop Lee
School of International Engineering and Science
CHONBUK NATIONAL UNIVERSITY

Outline

- Structures
- File Input & Output

- C arrays allow you to define type of variables that can <u>hold several data items of</u> the same kind.
- **C** structure is another <u>user defined data type</u> available in C programming, which allows you to <u>combine data items of **different kinds**</u>.
- Generally, structures are used to <u>represent a record</u>.
- Suppose you want to keep track of your books in a library.
- You might want to track the following attributes about each book:
 - Title
 - Author
 - Subject
 - Book ID

```
struct Books
{
    char title[50];
    char author[50];
    char subject[100];
    int book id;
} book;
```



Defining a structure

- To define a structure, you must use the struct statement.
- The **struct statement** defines a <u>new data type (structure type)</u>, with more than one member for your program.
- The format of the **struct statement** is this:

```
struct [structure tag]
{
   member definition;
   member definition;
   ...
   member definition;
} [one or more structure variables];
```



```
char title[50];
char author[50];
char subject[100];
int book id;
book;
```

- The structure tag is <u>optional</u>.
- Each member definition is a <u>normal variable or array definition</u>.
- One or more structure variables is optional.
- Note that there is the final semicolon at the end of the structure's definition.

Defining a structure

- To define a structure, you must use the struct statement.
- The **struct statement** defines a <u>new data type (structure type)</u>, with more than one member for your program.
- The format of the struct statement is this:

```
struct [structure tag]
{
   member definition;
   member definition;
   ...
   member definition;
} [one or more structure variables];
```

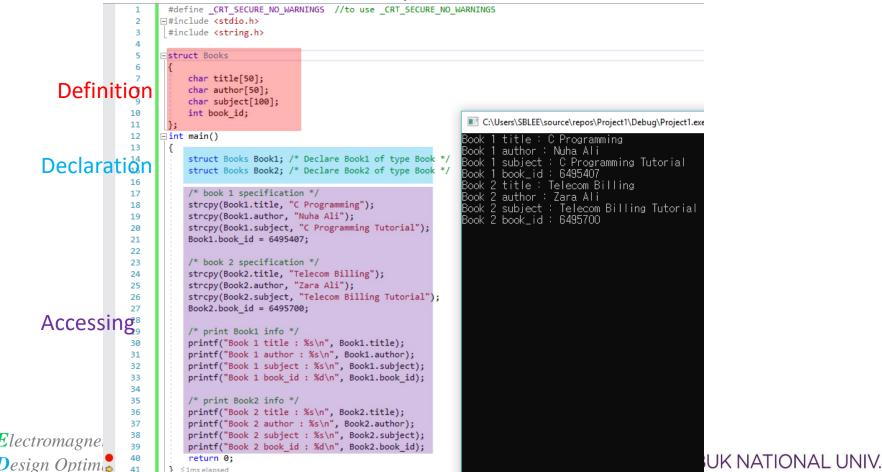


```
char title[50];
char author[50];
char subject[100];
int book id;
} book;
```

Declaration of the structure in functions

 The syntax of the declaration of the structure is the same to that of the variable except the struct.

- Accessing structure members
 - To access any member of a structure, you must use the member access operator (.).
 - The **member access operator** is coded as a **period** (.) between the <u>structure variable</u> <u>name</u> and the <u>structure member</u> that you wish to access.



- Structures as Functions Arguments
 - You can pass a structure as a function argument in very similar way as you pass any other variable or pointer.
 - You would access structure variables in the similar way as you have accessed in the

```
#define _CRT_SECURE_NO_WARNINGS //to use _CRT_SECURE_NO_WARNINGS
      above example:
                                                    ∃#include <stdio.h>
                                                    #include <string.h>
                                                    -struct Books
                                                        char title[50];
                              Definition
                                                        char author[50];
                                                        char subject[100];
                                                        int book id;
                                                                                                            C:\Users\SBLEE\source\repos\Project1\Debug\Project*
                                             11
                                                    /* function declaration */
                                                                                                                title: C Programming
                                                    void printBook(struct Books book);
                                                                                                                author : Nuha Ali
                                                    □int main()
                                                                                                                subject : C Programming Tutorial
                                                                                                                book_id: 6495407
                                                        struct Books Book1; /* Declare Book1 of type Book */
                              Declaration
                                                        struct Books Book2; /* Declare Book2 of type Book */
                                                        /* book 1 specification */
                                                                                                               subject : Telecom Billing Tutorial
                                                        strcpy(Book1.title, "C Programming");
                                                                                                           Book book id : 6495700
                                                        strcpy(Book1.author, "Nuha Ali");
                                             20
                                                        strcpy(Book1.subject, "C Programming Tutorial");
                                                        Book1.book id = 6495407;
                                                        /* book 2 specification */
                              Accessing
                                                        strcpy(Book2.title, "Telecom Billing");
                                                        strcpy(Book2.author, "Zara Ali");
                                                        strcpy(Book2.subject, "Telecom Billing Tutorial");
                                             26
                                             27
                                                        Book2.book id = 6495700;
                                             28
                                                        /* print Book1 info */
                                             29
                                                        printBook(Book1);
                                             30
                                                        /* Print Book2 info */
                                             31
                                                        printBook(Book2);
                                             32
                                                         return 0;
                                             33
                                                     void printBook(struct Books book)
                                             34
                                             35
                                             36
                                                        printf("Book title : %s\n", book.title);
                                             37
                                                        printf("Book author : %s\n", book.author);
Electromagnetic Systems
                                             38
                                                        printf("Book subject : %s\n", book.subject);
                                             39
                                                        printf("Book book_id : %d\n", book.book_id);
Design Optimization LAB
```

Thank You