Programming with C and C++

CSC-101 (Lecture 26)

Dr. R. Balasubramanian
Professor
Department of Computer Science and Engineering
Mehta Family School of Data Science and Artificial Intelligence
Indian Institute of Technology Roorkee
Roorkee 247 667

bala@cs.iitr.ac.in
https://faculty.iitr.ac.in/cs/bala/

Structures in C

```c
#include<stdio.h>
#include <string.h>

struct employee
{    int id;
     char name[50];
     float salary;
} e1,e2;  //declaring e1 and e2 variables for structure

int main( )
{
   //store first employee information
   e1.id=1;
   strcpy(e1.name, "KL Rahul");//copying string into char array
   e1.salary=5000000;

```

https://ideone.com/GHD7Jr

Structures in C

```
17      //store second employee information
18       e2.id=18;
19       strcpy(e2.name, "Virat");
20       e2.salary=10000000;
21
22       //printing first employee information
23       printf( "employee 1 id : %d\n", e1.id);
24       printf( "employee 1 name : %s\n", e1.name);
25       printf( "employee 1 salary : %.2f\n\n", e1.salary);
26
27       //printing second employee information
28       printf( "employee 2 id : %d\n", e2.id);
29       printf( "employee 2 name : %s\n", e2.name);
30       printf( "employee 2 salary : %.2f\n", e2.salary);
31       return 0;
32   }
33
```

Structures in C

```
Success #stdin #stdout 0s 5512KB
employee 1 id : 1
employee 1 name : KL Rahul
employee 1 salary : 5000000.00

employee 2 id : 18
employee 2 name : Virat
employee 2 salary : 10000000.00
```

Array of Structures

```
1  #include<stdio.h>
2  #include <string.h>
3
4      struct student{
5          int rollno;
6          char name[10];
7      };
8
9      int main(){
10     int i;
11     struct student st[3];
12     printf("Enter Records of 3 students");
13
```

https://ideone.com/qhBhmi

Array of Structures

```
14      for(i=0;i<3;i++){
15          printf("\nEnter Rollno:");
16          scanf("%d",&st[i].rollno);
17          printf("\nEnter Name:");
18          scanf("%s",&st[i].name);
19      }
20
21      printf("\nStudent Information List:");
22
23      for(i=0;i<3;i++){
24          printf("\nRollno:%d, Name:%s",st[i].rollno,st[i].name);
25      }
26      return 0;
27  }
28
```

stdin

```
1
Ravi
2
Rahul
3
Rohit
```

```
Student Information List:
Rollno:1, Name:Ravi
Rollno:2, Name:Rahul
Rollno:3, Name:Rohit
```

Pointers to Structures

```
struct Books Book1;

struct Books *struct_pointer;

struct_pointer = &Book1;

struct_pointer->title;
```

Structure Pointer in C

https://ideone.com/A3cVTj

```c
#include <stdio.h>
#include <string.h>

struct Course
{
    char cou_name[40];
    int cou_id;
    char cou_duration[30];
    char cou_type[30];
};

int main()
{
    struct Course cou;
    struct Course *ptr;
    ptr = &cou;  // Assign the address of 'cou' to the pointer

```

Structure Pointer in C

```
18      strcpy(cou.cou_name, "Computer Science and Engineering");
19      cou.cou_id = 101;
20      strcpy(cou.cou_duration, "4 Months");
21      strcpy(cou.cou_type, "Theory and Practical");
22
23      printf("Course Name: %s\n", (*ptr).cou_name);
24      printf("Course ID: %d\n", (*ptr).cou_id);
25      printf("Duration of the Course: %s\n", (*ptr).cou_duration);
26      printf("Type of the Course: %s\n", (*ptr).cou_type);
27
28      return 0;
29  }
30
```

stdout

```
Course Name: Computer Science and Engineering
Course ID: 101
Duration of the Course: 4 Months
Type of the Course: Theory and Practical
```

Structure Pointer in C

https://ideone.com/XoZ8H4

```c
#include <stdio.h>
#include <string.h>

struct Course
{
    char cou_name[40];
    int cou_id;
    char cou_duration[30];
    char cou_type[30];
};

int main()
{
    struct Course cou;
    struct Course *ptr;
    ptr = &cou;  // Assign the address of 'cou' to the pointer

```

Structure Pointer in C

```
18      strcpy(cou.cou_name, "Computer Science and Engineering");
19      cou.cou_id = 1201;
20      strcpy(cou.cou_duration, "6 Months");
21      strcpy(cou.cou_type, "Multiple Choice Question");
22
23      //Print the details of the Course
24      printf("Course Name: %s\n", ptr->cou_name);
25      printf("Course ID: %d\n", ptr->cou_id);
26      printf("Duration of the Course: %s\n", ptr->cou_duration);
27      printf("Type of the Course: %s\n", ptr->cou_type);
28
29      return 0;
30  }
31
```

stdout

```
Course Name: Computer Science and Engineering
Course ID: 101
Duration of the Course: 4 Months
Type of the Course: Theory and Practical
```

Addition of 2 Distances

https://ideone.com/3Bz7fU

```c
#include <stdio.h>
// Structure to represent distance
typedef struct {
    int feet;
    int inches;
} Distance;
// Function to add two distances
Distance addDistances(Distance d1, Distance d2) {
    Distance result;
    result.feet = d1.feet + d2.feet;
    result.inches = d1.inches + d2.inches;

    // Adjust inches to be less than 12
    if (result.inches >= 12) {
        result.inches -= 12;
        result.feet++;
    }

    return result;
}

```

```c
int main() {
    Distance distance1, distance2, sum;

    // Input first distance
    printf("Enter first distance (feet inches): ");
    scanf("%d %d", &distance1.feet, &distance1.inches);

    // Input second distance
    printf("Enter second distance (feet inches): ");
    scanf("%d %d", &distance2.feet, &distance2.inches);

    // Add the distances
    sum = addDistances(distance1, distance2);

    // Display the sum
    printf("Sum of distances: %d feet %d inches\n", sum.feet, sum.inches);

    return 0;
}
```

stdin

```
7 8
6 7
```

stdout

```
Enter first distance (feet inches): 7 8
Enter second distance (feet inches): 6 7
Sum of distances: 14 feet 3 inches
```

Pointers to Structures

https://ideone.com/61hGzX

```
1  #include <stdio.h>
2  #include <string.h>
3
4  struct Books {
5     char  title[60];
6     char  author[50];
7     char  subject[100];
8     int   book_id;
9  };
10
11 /* function declaration */
12 void printBook( struct Books *book );
13
14 int main( ) {
15
16    struct Books Book1;        /* Declare Book1 of type Book */
17    struct Books Book2;        /* Declare Book2 of type Book */
18
```

```
19      /* book 1 specification */
20      strcpy( Book1.title, "The C Programming Language");
21      strcpy( Book1.author, "Brian Kernighan");
22      strcpy( Book1.subject, "Computer Science");
23      Book1.book_id = 1234567;
24
25      /* book 2 specification */
26      strcpy( Book2.title, "Data Structures Algorithms and Applications in C++");
27      strcpy( Book2.author, "Sartaj Sahni");
28      strcpy( Book2.subject, "Computer Science");
29      Book2.book_id = 7654321;
30
```

```
31      /* print Book1 info by passing address of Book1 */
32      printBook( &Book1 );
33
34      /* print Book2 info by passing address of Book2 */
35      printBook( &Book2 );
36
37      return 0;
38  }
39
40  void printBook( struct Books *book ) {
41
42      printf( "Book title : %s\n", book->title);
43      printf( "Book author : %s\n", book->author);
44      printf( "Book subject : %s\n", book->subject);
45      printf( "Book book_id : %d\n\n", book->book_id);
46  }
47
```

Success #stdin #stdout 0.01s 5436KB

```
Book title : The C Programming Language
Book author : Brian Kernighan
Book subject : Computer Science
Book book_id : 1234567

Book title : Data Structures Algorithms and Applications in C++
Book author : Sartaj Sahni
Book subject : Computer Science
Book book_id : 7654321
```

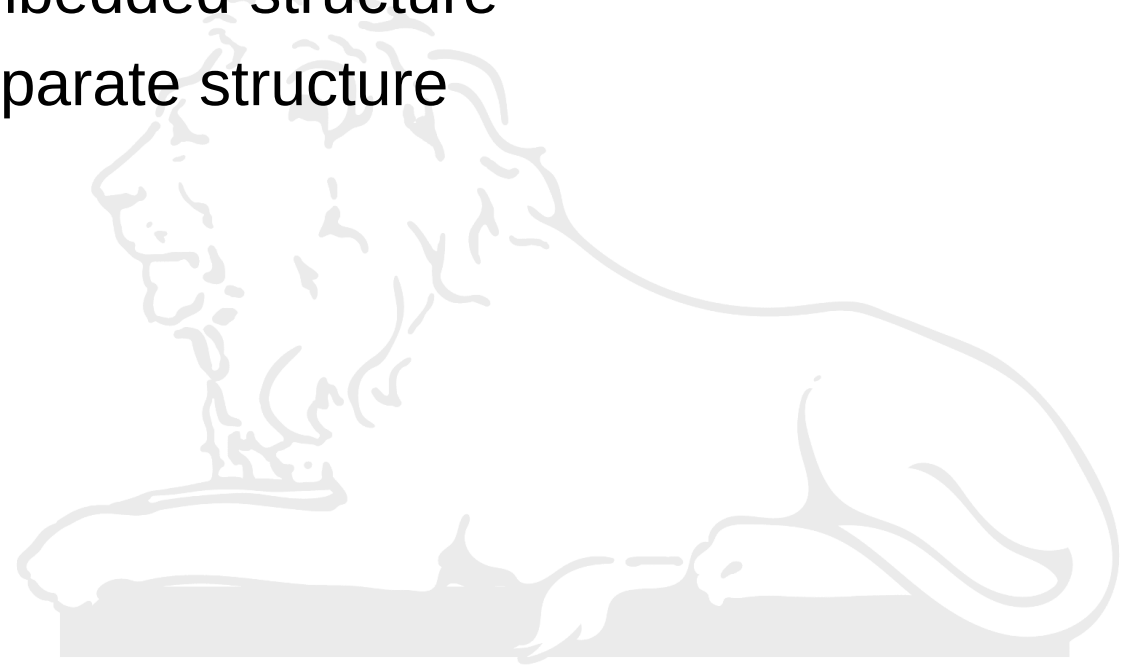
Nested Structure in C

- C provides us the feature of nesting one structure within another structure by using which, complex data types are created.

```
struct address
{
    char city[20];
    int pin;
    char phone[14];
};
struct employee
{
    char name[20];
    struct address add;
};
```

Nested Structure in C

- The structure can be nested in the following ways.
  - By Embedded structure
  - By Separate structure



Embedded structure

```
struct Employee
{
    int id;
    char name[20];
    struct Date
     {
       int dd;
       int mm;
       int yyyy;
     }doj;
}emp1;
```

Embedded structure

```
1   #include <stdio.h>
2   #include <string.h>
3
4   struct Employee
5   {
6       int id;
7       char name[20];
8       struct Date
9        {
10         int dd;
11         int mm;
12         int yyyy;
13       }doj;
14  }e1;
15
```

https://ideone.com/vi7Oi8

Embedded structure

```
16. int main( )
17. {
18.    //storing employee information
19.    e1.id=18;
20.    strcpy(e1.name, "Virat");//copying string into char array
21.    e1.doj.dd=11;
22.    e1.doj.mm=10;
23.    e1.doj.yyyy=2023;
24.
25.    //printing first employee information
26.    printf( "employee id : %d\n", e1.id);
27.    printf( "employee name : %s\n", e1.name);
28.    printf( "employee date of joining (dd/mm/yyyy) : %d/%d/%d\n", e1.doj.dd,e1.doj.mm,
    e1.doj.yyyy);
29.    return 0;
30. }
```

stdout

```
employee id : 18
employee name : Virat
employee date of joining (dd/mm/yyyy) : 11/10/2023
```

Separate structure Example

```
struct Date
{
    int dd;
    int mm;
    int yyyy;
};
struct Employee
{
    int id;
    char name[20];
    struct Date doj;
}emp1;
```

Separate structure

```
1   #include<stdio.h>
2
3   struct address
4   {
5       char city[20];
6       int pin;
7       char phone[14];
8   };
9
10  struct employee
11  {
12      char name[20];
13      struct address add;
14  };
15
```

https://ideone.com/fxcwMt

Separate structure

```
16.  void main ()
17.  {
18.      struct employee emp;
19.      printf("Enter employee information?\n");
20.      scanf("%s %s %d %s",emp.name,emp.add.city, &emp.add.pin, emp.add.phone);
21.      printf("Printing the employee information....\n");
22.      printf("name: %s\nCity: %s\nPincode: %d\nPhone: %s",emp.name,emp.add.city,emp.ad
     d.pin,emp.add.phone);
23.  }
```

stdin

```
Rohit
Nagpur
440001
9876543210
```

stdout

```
Enter employee information?
Printing the employee information....
name: Rohit
City: Nagpur
Pincode: 440001
Phone: 9876543210
```

```
struct employee emp;
printf("Enter employee information?\n");
scanf("%s %s %d %s",emp.name,emp.add.city, &emp.add.pin, emp.add.phone);
printf("Printing the employee information....\n");

printf("name: %s\nCity: %s\nPincode: %d\nPhone: %s",emp.name,emp.add.city,
emp.add.pin,emp.add.phone);
```

Self Referential Structure

```
1   #include <stdio.h>
2   #include <stdlib.h>
3
4   struct Node {
5       int data;
6       struct Node* next;
7   };
8
9   void display(struct Node* temp)
10  {
11      while (temp) {
12      printf(" %d ", temp->data);
13      temp= temp->next;
14      }
15  }
16
```

https://ideone.com/8vguyR

Self Referential Structure

```
17  void main()
18  {
19      // assign each node a null value to avoid any refrence error
20      struct Node* head =NULL;
21      struct Node* second_node = NULL;
22      struct Node* third_node = NULL;
23
24      // defining three nodes
25      head = (struct Node*)malloc(sizeof(struct Node));
26      second_node = (struct Node*)malloc(sizeof(struct Node));
27      third_node = (struct Node*)malloc(sizeof(struct Node));
28
29      head->data = 1000; // assign data in first node
30      head->next = second_node; // Link first node with second
31
32      second_node->data = 2000; // assign data to second node
33      second_node->next = third_node;
```

Self Referential Structure

```
34
35      third_node->data = 3000; // assign data to third node
36      third_node->next = NULL;
37
38      // calling the function to display value
39      display(head);
40
41      free(head);
42      free(second_node);
43      free(third_node);
44  }
45
```

stdout

```
1000   2000   3000
```

Thank You!