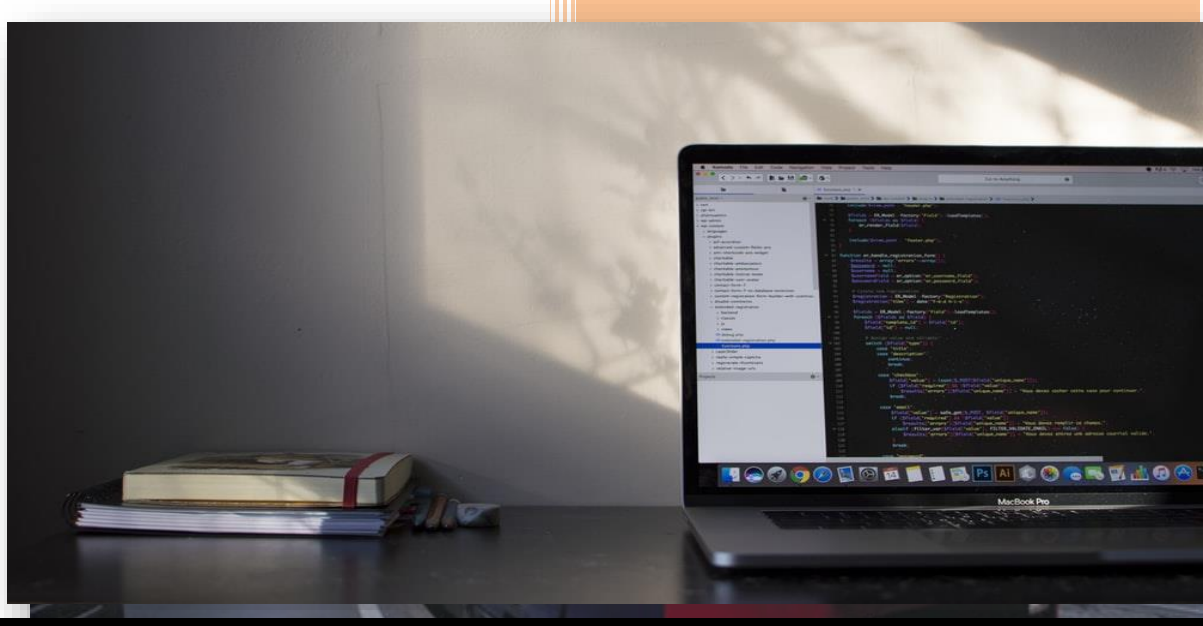
ETEC Center

Build Your IT Skill

2020



Structure of C Programming

ណែនាំអោយស្គាល់ពី Structure របស់ C Programming

1. ដូចម្តេចទៅដែលហៅថា Structure?

Structure គឺសំដៅលើការប្រមូលផ្តុំនូវទិន្នន័យ ឬ ពត៌មានផ្សេងៗ ដាក់ជាបណ្តុំមួយក្រោមឈ្មោះតែមួយ ដោយប្រើប្រាស់នូវ keyword struct ដើម្បីបង្កើតវាឡើង។ Structure វាអាចប្រើប្រាស់នូវទិន្នន័យទាំងនោះបានត្រូវប្រាស់តាមរយៈការ Object របស់ Structure ដែលវាអាចបង្កើតចេញពី ឈ្មោះ structure តែម្តង ។

<u>ទំរង់ទ្ធទៅរបស់ Structure</u>

```
1  struct structureName
2  {
3      dataType member1;
4      dataType member2;
5      ...
6  };
```

ឧទាហរណ៏៖ ចូរបង្កើត Structure មួយដែលមានឈ្មោះ Person ដូចខាងក្រោម៖

```
1
2  struct Person
3  {
4      char name[50];
5      int id;
6      float salary;
7      char sex;
8  };
9
10
```

2. ការប្រើប្រាស់នូវទិន្នន័យរបស់ Structure

ដើម្បីអាចប្រើប្រាស់នូវ ទិន្នន័យដែលមាននៅក្នុង Structure អ្នកត្រូវបង្កើតនូវ Object របស់ Structureដោយប្រើប្រាស់នូវ Keyword struct ជាដំបូងសិន បន្ទាប់មក ត្រូវ យក Objectទាំងនោះទៅ dot(.) ជាមួយនឹង ទិន្នន័យក្នុង Structure ជាការស្រេច។

ឧទាហរណ៏ ១៖

```c

struct Person
{
    char name[50];
    int id;
    float salary;
    char sex;
};
int main()
{
  struct Person person1, person2;

    printf("Input ID=");
    scanf("%d",&person1.id);
    .........

  return 0;
}
```

ឬ

```c
#include<stdio.h>
struct Person
{
    char name[50];
    int id;
    float salary;
    char sex;
}person1, person2;
int main()
{

    printf("Input ID=");
    scanf("%d",&person1.id);
    .........

  return 0;
}
```

ឧទាហរណ៏ ១

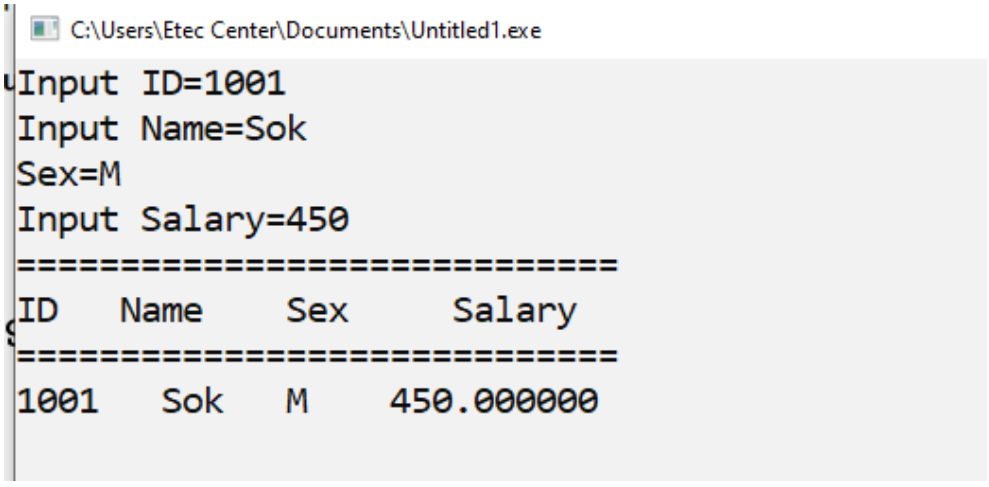
```c
#include<stdio.h>
struct Person
{
    char name[50];
    int id;
    float salary;
    char sex;
}person1;
int main()
{

    printf("Input ID=");
    scanf("%d",&person1.id);
    printf("Input Name=");
    fflush(stdin);
    gets(person1.name);
    printf("Sex=");
    scanf("%c",&person1.sex);
    printf("Input Salary=");
    scanf("%f",&person1.salary);
    printf("==============================\n");
    printf("ID   Name    Sex     Salary\n");
    printf("==============================\n");
    printf("%d   %s   %c    %f\n",person1.id,person1.name,person1.sex,person1.salary);

  return 0;
}

```

លទ្ធផលទទួលបាន៖



ឧទាហរណ៏ ២៖

```c
#include <stdio.h>
#include <string.h>
struct Books {
   char  title[50];
   char  author[50];
   char  subject[100];
   int   book_id;
};
int main( ) {

   struct Books Book1;        /* Declare Book1 of type Book */
   struct Books Book2;        /* Declare Book2 of type Book */
   /* book 1 specification */
   strcpy( Book1.title, "C Programming");
   strcpy( Book1.author, "Heng Pheakna");
   strcpy( Book1.subject, "Basic C Programming");
   Book1.book_id = 1001;
   /* book 2 specification */
   strcpy( Book2.title, "C++ Programming");
   strcpy( Book2.author, "Thai Symeng");
   strcpy( Book2.subject, "Basic C++ Programming");
   Book2.book_id = 1002;
   /* print Book1 info */
   printf( "Book 1 title : %s\n", Book1.title);
   printf( "Book 1 author : %s\n", Book1.author);
   printf( "Book 1 subject : %s\n", Book1.subject);
   printf( "Book 1 book_id : %d\n", Book1.book_id);

   /* print Book2 info */
   printf( "Book 2 title : %s\n", Book2.title);
   printf( "Book 2 author : %s\n", Book2.author);
   printf( "Book 2 subject : %s\n", Book2.subject);
   printf( "Book 2 book_id : %d\n", Book2.book_id);

   return 0;
}
```

លទ្ធផលទទួលបាន៖

```
C:\Users\Etec Center\Documents\Untitled1.exe
Book 1 title : C Programming
Book 1 author : Heng Pheakna
Book 1 subject : Basic C Programming
Book 1 book_id : 1001
Book 2 title : C++ Programming
Book 2 author : Thai Symeng
Book 2 subject : Basic C++ Programming
Book 2 book_id : 1002
```

3. ការប្រើប្រាស់នូវ Function ជាមួយនិង Structure

ក្នុងការប្រើប្រាស់នូវ Structure អ្នកក៏អាចយក Statement Codeណាដែលសុគស្មាញ ទៅ សរសេរ ជា Function បានផងដែរ តាមទំរង់ Return function និង Return function។

ឧទាហរណ៏ ១៖

```c
#include <stdio.h>
#include <string.h>
struct Books {
   char  title[50];
   char  author[50];
   char  subject[100];
   int   book_id;
};

/* function declaration */
void Output( struct Books book );

int main( ) {

   struct Books Book1;        /* Declare Book1 of type Book */
   struct Books Book2;        /* Declare Book2 of type Book */

   /* book 1 specification */
   strcpy( Book1.title, "C Programming");
   strcpy( Book1.author, "Heng Pheakna");
   strcpy( Book1.subject, "C Programming Tutorial");
   Book1.book_id = 1001;

   /* book 2 specification */
   strcpy( Book2.title, "C++ Programming");
   strcpy( Book2.author, "Heng Pheakna");
   strcpy( Book2.subject, "Basic C++ Programming");
   Book2.book_id = 1002;

   /* print Book1 info */
   Output( Book1 );

   /* Print Book2 info */
   Output( Book2 );

   return 0;
}
void Output( struct Books book ) {

   printf( "Book title : %s\n", book.title);
   printf( "Book author : %s\n", book.author);
   printf( "Book subject : %s\n", book.subject);
   printf( "Book book_id : %d\n", book.book_id);
}
```

លទ្ធផលទទួលបាន៖

```
Book title : C Programming
Book author : Heng Pheakna
Book subject : C Programming Tutorial
Book book_id : 1001
Book title : C++ Programming
Book author : Heng Pheakna
Book subject : Basic C++ Programming
Book book_id : 1002

--------------------------------
```

ឧទាហរណ៏ ២៖ ការបង្កើតនូវ function លក្ខណៈជា return structure Object

```c
#include <stdio.h>
struct student {
  char firstname[64];
  char lastname[64];
  char id[64];
  int score;
};
// function declaration
struct student getDetail(void);
void displayDetail(struct student std);
int main(void) {
  // creating a student structure array variable
  struct student stdArr[3];
  // other variables
  int i;
  // taking user input
  for (i = 0; i < 3; i++) {
    printf("Enter detail of student #%d\n", (i+1));
    stdArr[i] = getDetail();
  }
  // output
  for (i = 0; i < 3; i++) {
    printf("\nStudent #%d Detail:\n", (i+1));
    displayDetail(stdArr[i]);
  }
  return 0;
}
```

```c
//Function return structure object
struct student getDetail(void) {
  // temp structure variable
  struct student std;
  printf("Enter First Name: ");
  scanf("%s", std.firstname);
  printf("Enter Last Name: ");
  scanf("%s", std.lastname);
  printf("Enter ID: ");
  scanf("%s", std.id);
  printf("Enter Score: ");
  scanf("%d", &std.score);
  return std;
}
void displayDetail(struct student std) {
  printf("Firstname: %s\n", std.firstname);
  printf("Lastname: %s\n", std.lastname);
  printf("ID: %s\n", std.id);
  printf("Score: %d\n", std.score);
}
```

លទ្ធផលទទួលបាន៖

```
C:\Users\Etec Center\Documents\Untitled1.exe
Enter Last Name: Lin
Enter ID: 1001
Enter Score: 78
Enter detail of student #3
Enter First Name: Keo
Enter Last Name: vanna
Enter ID: 1003
Enter Score: 67

Student #1 Detail:
Firstname: Sok
Lastname: Dara
ID: 1002
Score: 76

Student #2 Detail:
Firstname: Jame
Lastname: Lin
ID: 1001
Score: 78

Student #3 Detail:
Firstname: Keo
Lastname: vanna
ID: 1003
Score: 67
```

4. ការប្រើប្រាស់នូវ Pointer ជាមួយនិង Structure

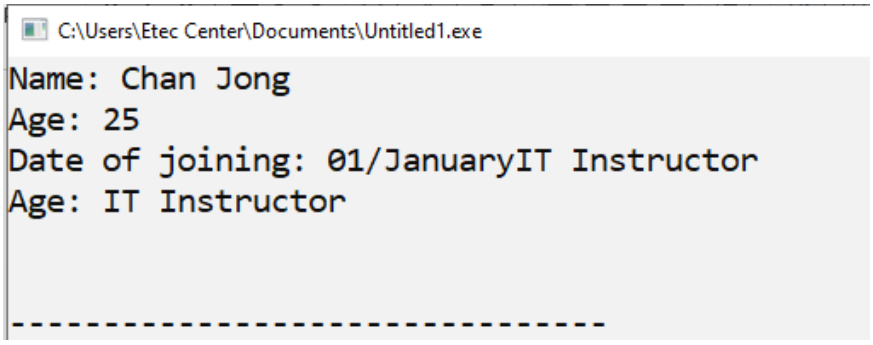
ការយក Pointer មកប្រើប្រាស់ជាមួយនិង Structure មានន័យថាអ្នកអាចធ្វើការបង្កើតនូវ Object របស់ Structure ជាទំនង់ Pointer ពោលគឺអាចអោយ pointer ចង្អុលទៅកាន់ Object Structure ផ្សេងទៀត ឬ អាចបង្កើត Object Pointer ជាទំនង Dynamic Memory Allocation(DMA) ។

ឧទាហរណ៏ ១៖

```c
#include<stdio.h>
struct employee
{
    char name[20];
    int age;
    char doj[10]; // date of joining
    char designation[20];
};
void print_struct(struct employee *);
int main()
{
    struct employee dev = {"Chan Jong", 25, "01/January/2020", "IT Instructor"};
    struct employee *p;
    p=&dev;
    printf("Name: %s\n", p->name);
    printf("Age: %d\n", p->age);
    printf("Date of joining: %s\n", p->doj);
    printf("Age: %s\n", p->designation);
    printf("\n");
    return 0;
}

```

លទ្ធផលទទួលបាន៖



ឧទាហរណ៏ ២៖

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

struct Books {
    int code;
    char  title[20];
    int qty;
    float price;
};
int main()
{
    struct Books *p;
    p=(struct Books *)malloc(sizeof(struct Books));
  //Input
        printf("Input Code=");scanf("%d",&p->code);
        fflush(stdin);
        printf("Input Title=");gets(p->title);
        printf("Input Quantity=");scanf("%d",&p->qty);
        printf("Input Unit Price=");scanf("%f",&p->price);
 //output
        printf("Code  Name   Quantity   Price\n");
           printf("%d  ",p->code);
           printf("%s  ",p->title);
           printf("%d  ",p->qty);
           printf("%f  ",p->price);
}
```

លទ្ធផលទទួលបាន៖

```
C:\Users\Etec Center\Documents\Untitled1.exe
Input Code=1001
Input Title=C Programming
Input Quantity=2
Input Unit Price=1.67
Code  Name   Quantity   Price
1001  C Programming  2  1.670000
--------------------------------
Process exited after 10.48 seconds with return value 10
Press any key to continue . . .
```

ឧទាហរណ៏ ៣៖

```c
/*C program to read and print the N student
details using structure and Dynamic Memory Allocation.*/
#include <stdio.h>
#include <stdlib.h>
/*structure declaration*/
struct student
{
    char name[30];
    int roll;
    float score;
};
int main()
{
    struct student *pstd;
    int n,i;
    printf("Enter total number of elements: ");
    scanf("%d",&n);
    /*Allocate memory dynamically for n objetcs*/
    pstd=(struct student*)malloc(n*sizeof(struct student));

    if(pstd==NULL)
    {
        printf("Insufficient Memory, Exiting... \n");
        return 0;
    }
    /*read and print details*/
    for(i=0; i<n; i++)
    {
        printf("\nEnter detail of student [%3d]:\n",i+1);
        printf("Enter name: ");
        scanf(" "); /*clear input buffer*/
        gets((pstd+i)->name);
        printf("Enter roll number: ");
        scanf("%d",&(pstd+i)->roll);
        printf("Enter Score: ");
        scanf("%f",&(pstd+i)->score);
    }
    printf("\nEntered details are:\n");
    for(i=0; i<n; i++)
    {
        printf("%30s \t %5d \t %.2f\n",(pstd+i)->name,
                                (pstd+i)->roll,(pstd+i)->score);
    }

    free(pstd);
    return 0;
}
```

លទ្ធផលទទួលបាន៖

```
C:\Users\Etec Center\Documents\Untitled2.exe
Enter detail of student [  2]:
Enter name: Ly Vanna
Enter roll number: 1002
Enter Score: 89.7

Entered details are:
                    Sok Chan    1001   67.90
                    Ly Vanna    1002   89.70

--------------------------------
Process exited after 40.62 seconds with return value 0
Press any key to continue . . .
```

លំហាត់អនុវត្តន៍

គេមាន Structure មួយដូចខាងក្រោម៖

```
 5
 6  struct Employee {
 7      int code;
 8      char name[20];
 9      char sex;
10      float salary;
11  };
12
```

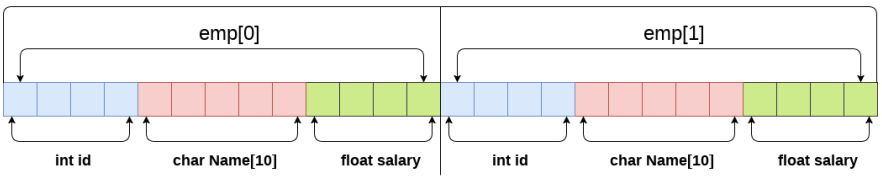
ចូរបង្កើតឆ្លូវកម្មវិធីតាម Menu ដូចខាងក្រោម៖

```
>>>>>>>>>>>> Menu <<<<<<<<<<<<<<<
   >>> 1. Input
   >>> 2. Output
   >>> 3. Search
   >>> 6. Exit
 Please Choose One(1-6)=___
```

5. ការប្រើប្រាស់នូវ Array ជាមួយនិង Structure

ក្នុង Structure អ្នកក៏អាចបង្កើតនូវ Object Structure ជា Array បានផង់ដែរ ។ អ្នកត្រូវចាំថា 1 Object index ស្មើរនិងចំនួន record ដែលមានក្នុង Structure។ ជាឧទាហរណ៏ បើក្នុង Structure របស់អ្នកមាន Data ចំនួន ៣នោះ Object របស់ វាក៏ផ្ទុកចំនួនដល់ទៅ ៣ផង់ដែរ។

Array of structures




```
struct employee
{
    int id;
    char name[5];
    float salary;
};
struct employee emp[2];
```

sizeof (emp) = 4 + 5 + 4 = 13 bytes

sizeof (emp[2]) = 26 bytes

ឧទាហរណ៏ ១៖ ចូរបង្កើតនូវ Structure មួយឈ្មោះ Books ដែលផ្ទុកទិន្នន័យដូចជា char title[50], char author[50], char subject[100] និង int book_id; និងអាចអោយគេបញ្ចូលពត៏មាន និងបង្ហាញពត៏មាន។

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

struct Books {
   char  title[50];
   char  author[50];
   char  subject[100];
   int   book_id;
};

int main()
{
    char op;
    int i,n,j;
    int b=0;
do{ system("cls");
    struct Books book[20];
    printf("......Menu Choice.....\n");
    printf("a- Input\n");
    printf("b- Output\n");
    printf("c- Search\n");
    printf("d- Update\n");
    printf("e- Sort\n");
    printf("Please Choose One=");
    scanf("%c",&op);
```

```c
    switch(op)
    {
        case 'a':
        case 'A':{
                printf("Input Number of Students=");scanf("%d",&n);
                for(i=0;i<n;i++)
                {
                  printf("........Book Record %d........\n",i+1);
                  printf("Input Code=");
                  scanf("%d",&book[i].book_id);
                  fflush(stdin);
                  printf("Input Title=");
                  gets(book[i].title);
                  fflush(stdin);
                  printf("Input Author=");
                  gets(book[i].author);
                  fflush(stdin);
                  printf("Input Subjec=");
                  gets(book[i].subject);
                }

        }   break;
        case 'b':
        case 'B':{
               printf("Book_ID     Title     Author     Subject\n");
               for(i=0;i<n;i++)
               {
                  printf("%d   %s   %s    %s\n",book[i].book_id,book[i].title,book[i].author,book[i].subject);
               }
    }break;
```

```c
    case 'c':
    case 'C':{
          b=0;
          int sid;
           printf("Input ID to Search=");
          scanf("%d",&sid);
          for(i=0;i<n;i++)
            {
                if(book[i].book_id==sid)
                {
                  printf("%d   %s   %s    %s\n",book[i].book_id,book[i].title,book[i].author,book[i].subject);
                  printf("Search found\n");
                  b=1;
                  break;
                  }
            }
         if(b==0) printf("Search not found\n");
    }break;
```

```
    case 'D':
      case 'd':{
            char ntitle[20],nauthor[20],nsub[20];
             int sid;
              b=0;
            printf("Input ID to Search=");
            scanf("%d",&sid);
            for(i=0;i<n;i++)
              {
                 if(book[i].book_id==sid)
                 {
                   printf("%d   %s   %s    %s\n",book[i].book_id,book[i].title,book[i].author,book[i].subject);
                   printf("Input New Titlt=");
                   gets(ntitle);
                   printf("Input New Author=");
                   gets(nauthor);
                   printf("Input New Subject=");
                   gets(nsub);
                   strcpy(book[i].title,ntitle);
                   strcpy(book[i].author,nauthor);
                   strcpy(book[i].subject,nsub);
                   b=1;
                   break;
                   }
              }
           if(b==0) printf("Search not found\n");
      }break;
```

```
       case 'e':
      case 'E':{
              struct Books tbook;
              for(i=0;i<n;i++)
              {
                  for(j=i+1;j<n;j++)
                  {
                      if(strcmp(book[i].title,book[j].title)>0)
                      {
                          tbook=book[i];
                          book[i]=book[j];
                          book[j]=tbook;
                       }
                  }
              }

      }break;
}
printf("Press Enter to Continue.....!\n");
}while(getch()==13);
}
```

លំហាត់អនុវត្តន៍

១) ចូរបង្កើតនូវ Structure ឈ្មោះ Dictionary មួយដែលអាចអោយគេផ្ទុកពត៌មានដូចជា Word(String), Speech(String) និង Description(String) បន្ទាប់មកបង្កើតនូវ Object Array មួយដែលអាចផ្ទុកនូវពាក្យចំនួន N Record ហើយបង្ហាញចេញមកក្រៅវិញ និង Search ពាក្យទាំងនោះបាន?

```
1   #include <stdio.h>
2   #include <string.h>
3
4   struct Dictionarys {
5       string word;
6       string speec;
7       string description;
8   };
9
10  int main()
11  {
12      struct Dictionarys dic[20];
13      int n;
14
15
16
17
18
19      return 0;
20  }
21
22
23
```

6. ការបង្កើតនូវ Structure មួយក្នុង Structure មួយទៀត/ Nested Structure

នៅក្នុងភាសារ C Programming អ្នកអាចធ្វើការបង្កើតនូវ ទំរង់ Structure ក្នុង Structure មួយទៀតបាន ពោលStructure មួយក្នុង Structure មួយទៀត ។

```
3   struct Employee
4   {
5       char ename[20];
6       int ssn;
7       float salary;
8       struct date
9           {
10          int date;
11          int month;
12          int year;
13          }doj;
14  }emp = {"Pritesh",1000,1000.50,{22,6,1990}};
```

ឧទាហរណ៏ ១៖ (Embed Structure)

```
1  /*C program to demonstrate example of nested structure*/
2  #include <stdio.h>
3  struct student{
4      char name[30];
5      int rollNo;
6
7      struct dateOfBirth{
8          int dd;
9          int mm;
10         int yy;
11     }DOB;   /*created structure varoable DOB*/
12 };
13 int main()
14 {
15     struct student std;
16
17     printf("Enter name: ");          gets(std.name);
18     printf("Enter roll number: ");   scanf("%d",&std.rollNo);
19     printf("Enter Date of Birth [DD MM YY] format: ");
20     scanf("%d%d%d",&std.DOB.dd,&std.DOB.mm,&std.DOB.yy);
21     printf("\nName : %s \nRollNo : %d \nDate of birth : %02d/%02d/%02d\n",
22                                                  std.name,std.rollNo,
23                                          std.DOB.dd,std.DOB.mm,std.DOB.yy);
24     return 0;
25 }
```

លទ្ធផលទទួលបាន៖

```
C:\Users\Etec Center\Documents\DemoCStruct.exe
Enter name: Sok Dara
Enter roll number: 1001
Enter Date of Birth [DD MM YY] format: 01 02 1990

Name : Sok Dara
RollNo : 1001
Date of birth : 01/02/1990
```

ឧទាហរណ៏ ២៖ (Separate Structure)

```c
#include <stdio.h>
#include <string.h>
struct student_college_detail
{
    int college_id;
    char college_name[50];
};
struct student_detail
{
    int id;
    char name[20];
    float score;
    // structure within structure
    struct student_college_detail clg_data;
}stu_data, *stu_data_ptr;
int main()
{
  struct student_detail stu_data = {1001, "Chan Vertey",
                                    90.5, 2019,
                                    "ETEC CENTER"};
    stu_data_ptr = &stu_data;
    printf(" Id is: %d \n", stu_data_ptr->id);
    printf(" Name is: %s \n", stu_data_ptr->name);
    printf(" Score is: %f \n\n",
                        stu_data_ptr->score);
    printf(" College Id is: %d \n",
                        stu_data_ptr->clg_data.college_id);
    printf(" College Name is: %s \n",
                         stu_data_ptr->clg_data.college_name);
    return 0;
}
```

លទ្ធផលទទួលបាន៖

```
C:\Users\Etec Center\Documents\DemoCStruct.exe
Id is: 1001
Name is: Chan Vertey
Score is: 90.500000

College Id is: 2019
College Name is: ETEC CENTER

--------------------------------
```

លំហាត់អនុវត្តន៍

១) គេមានឆ្នូវទំរង់ Structure ដូចខាងក្រោមចូរធ្វើការបង្កើតឆ្នូវ Object សំរាប់ប្រើប្រាស់ដូចខាងក្រោម៖

```
1  struct Employee
2  {
3     int id;
4     char name[20];
5     float salary;
6     struct Date
7      {
8        int dd;
9        int mm;
10       int yyyy;
11     }date1;
12     struct Address{
13       char city[20];
14       int pin;
15       char phone[14];
16     }add;
17
18 }Emp;
19
```

Good Luck!