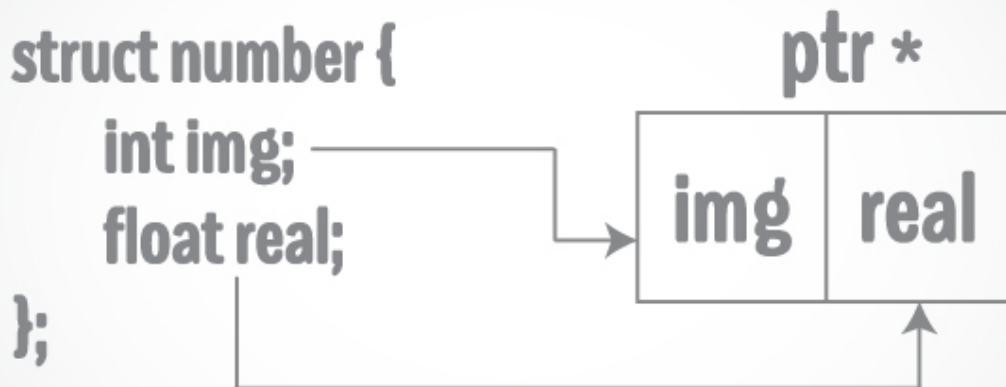


C Programming Structure and Pointer

In this article, you'll find relevant examples that will help you to work with pointers to access data within a structure.

**Download PDF**

To View PDF, Download Here



[Structures](#) can be created and accessed using [pointers](#). A pointer variable of a structure can be created as below:

```
struct name {  
    member1;  
    member2;  
    .  
    .  
};  
  
int main()
```



J

Here, the pointer variable of type `struct name` is created.

Accessing structure's member through pointer

A structure's member can be accessed through pointer in two ways:

1. Referencing pointer to another address to access memory
2. Using dynamic memory allocation

1. Referencing pointer to another address to access the memory

Consider an example to access structure's member through pointer.

```
#include <stdio.h>
typedef struct person
{
    int age;
    float weight;
};

int main()
{
    struct person *personPtr, person1;
    personPtr = &person1;           // Referencing pointer to memory address

    printf("Enter integer: ");
    scanf("%d",&(*personPtr).age);

    printf("Enter number: ");
    scanf("%f",&(*personPtr).weight);

    printf("Displaying: ");
    printf("%d%f",(*personPtr).age,(*personPtr).weight);

    return 0;
}
```

In this example, the pointer variable of type `struct person` is referenced to the address of `person1`. Then, only the structure member through pointer can be accessed.

Using `->` operator to access structure pointer member



```
(*personPtr).age is same as personPtr->age
(*personPtr).weight is same as personPtr->weight
```

2. Accessing structure member through pointer using dynamic memory allocation

To access structure member using pointers, memory can be allocated dynamically using [malloc\(\) function](#) defined under "stdlib.h" header file.

Syntax to use malloc()

```
ptr = (cast-type*) malloc(byte-size)
```

Example to use structure's member through pointer using malloc() function.

```
#include <stdio.h>
#include <stdlib.h>
struct person {
    int age;
    float weight;
    char name[30];
};

int main()
{
    struct person *ptr;
    int i, num;

    printf("Enter number of persons: ");
    scanf("%d", &num);

    ptr = (struct person*) malloc(num * sizeof(struct person));
    // Above statement allocates the memory for n structures with pointer personP

    for(i = 0; i < num; ++i)
    {
        printf("Enter name, age and weight of the person respectively:\n");
        scanf("%s%d%f", &(ptr+i)->name, &(ptr+i)->age, &(ptr+i)->weight);
    }

    printf("Displaying Infromation:\n");
    for(i = 0; i < num; ++i)
        printf("%s\t%d\t%.2f\n", (ptr+i)->name, (ptr+i)->age, (ptr+i)->weight);

    return 0;
}
```

Output

TUTORIAL

EXAMPLES



```
Adam
2
3.2
Enter name, age and weight of the person respectively:
Eve
6
2.3
Displaying Information:
Adam    2      3.20
Eve     6      2.30
```

Check out these examples to learn more:

- [Calculate Difference Between Two Time Periods](#)

PREVIOUS

[C STRUCTURES](#)

NEXT

[STRUCTURE & FUNCTION](#)



Download PDF

To View PDF, Download Here

C Programming

C Introduction



C Flow Control



C Functions



C Programming Arrays



[TUTORIAL](#)[EXAMPLES](#)

Structure And Union

[C Structure](#)[Structures & Pointers](#)[C Structure & Function](#)[C Programming Union](#)[Structure Examples](#)[C Programming Files](#)[Additional Topics](#)

Receive the latest tutorial to improve your programming skills

[Join](#)

Get Latest Updates on Programiz

[Subscribe](#)

[ABOUT](#)
[CONTACT](#)
[ADVERTISE](#)

Copyright © by Programiz | All rights reserved | [Privacy Policy](#)