# Unit :IV

# Array of pointers to structures

- Example 3:

```
#include <stdio.h>
struct TIME { int seconds; int minutes; int hours;
   };
void differenceBetweenTimePeriod(struct TIME
   t1, struct TIME t2, struct TIME *diff);
```

```
int main()
 { struct TIME startTime, stopTime, diff;

printf("Enter start time: \n");
printf("Enter hours, minutes and seconds respectively: ");
scanf("%d %d %d", &startTime.hours, &startTime.minutes,
    &startTime.seconds); printf("Enter stop time: \n");

printf("Enter hours, minutes and seconds respectively: ");
scanf("%d %d %d", &stopTime.hours, &stopTime.minutes,
    &stopTime.seconds);
```

```
// Calculate the difference between the start and stop time period.
    differenceBetweenTimePeriod(startTime, stopTime,
    &diff);


printf("\nTIME DIFFERENCE: %d:%d:%d - ",
    startTime.hours, startTime.minutes,
    startTime.seconds);


printf("%d:%d:%d ", stopTime.hours, stopTime.minutes,
    stopTime.seconds);


printf("= %d:%d:%d\n", diff.hours, diff.minutes,
    diff.seconds);
return 0; }
```

```
void differenceBetweenTimePeriod(struct TIME start,
    struct TIME stop, struct TIME *diff)
 { if(stop.seconds > start.seconds)
{ --start.minutes; start.seconds += 60; }

diff->seconds = start.seconds - stop.seconds;

if(stop.minutes > start.minutes)
{ --start.hours; start.minutes += 60; }

diff->minutes = start.minutes - stop.minutes;
 diff->hours = start.hours - stop.hours; }
```

- Example 4:

```
#include<stdio.h>

struct dog
{
   char name[10];
   char breed[10];
   int age;
   char color[10];
};

int main()
{
   struct dog  my_dog = {"tyke", "Bulldog", 5, "white"};
   struct dog *ptr_dog;
   ptr_dog = &my_dog;

   printf("Dog's name: %s\n", ptr_dog->name);
   printf("Dog's breed: %s\n", ptr_dog->breed);
   printf("Dog's age: %d\n", ptr_dog->age);
   printf("Dog's color: %s\n", ptr_dog->color);
```

```
    // changing the name of dog from tyke to jack
    strcpy(ptr_dog->name, "jack");

    // increasing age of dog by 1 year
    ptr_dog->age++;

    printf("Dog's new name is: %s\n", ptr_dog->name);
    printf("Dog's age is: %d\n", ptr_dog->age);

        return 0;
}
```

# Array of pointers to structures

- Like  array of integers, it is possible to declare array of pointers ,array of structure variables.
- And to use the array of structure variables efficiently, we use **pointers of structure type**.

# Array of pointers to structures

Example 1:

```
#include <stdio.h>
struct Book { char name[10]; int price; }
int main()
{ int i;
struct Book a; //Single structure variable
struct Book* ptr; //Pointer of Structure type
 ptr = &a;
struct Book b[3]; //Array of structure variables
struct Book* p; //Pointer of Structure type
 p = b ;//or p=&b[0];

printf("enter the details of books");
for(i=0;i<3;i++)
{
    printf("enter the name and price of book -%d",i+1);
    scanf("%s%d", &(p+i)->name, &(p+i)->price);
}
```

```
printf(“the details of books”);
for(i=0;i<3;i++)
{
   printf(“the name and price of book -%d”,i+1);
   printf(“%s%d”, (p+i)->name, (p+i)->price);
}

return 0;
}
```

Example 2:

```
#include <stdio.h>
 #include <stdlib.h>
struct person { int age; char name[30] };
 int main()
 { struct person *ptr;
int i, n;
 printf("Enter number of persons: ");
scanf("%d", &n);
 ptr = (struct person*) malloc(num * sizeof(struct person));
 for(i = 0; i < n; ++i)
 { printf("Enter first name and age respectively: "); scanf("%s%d", &(ptr+i)-> name,
     &(ptr+i)->age); }

printf("Displaying Information:\n");

for(i = 0; i < n; ++i)
printf("Name: %s\tAge: %d\n", (ptr+i)->name, (ptr+i)->age);

 return 0; }
```

- Example 3:

```
#include <stdio.h>
 struct student
 { char name[50]; int roll; float marks; } s[10],*ptr;

 int main()
 { int i;
ptr=s;
printf("Enter information of students:\n"); // storing information
for(i=0; i<10; ++i)
 {
(ptr+i)->roll = i+1;
 printf("\nFor roll number%d,\n“, (ptr+i)-> roll);
printf("Enter name: ");
scanf("%s“,&(ptr+i)-> name);
 printf("Enter marks: ");
scanf("%f",&(ptr+i)-> marks);
printf("\n"); }
```

```
printf("Displaying Information:\n\n");
// displaying information
for(i=0; i<10; ++i)
 { printf("\nRoll number: %d\n", (ptr+i)-> roll);
printf("Name: ");
puts((ptr+i)-> name);
 printf("Marks: %f", (ptr+i)-> marks);
printf("\n"); }
return 0; }
```