Pointer to array

1. Int arr[10];
2. int *p[10]=&arr;//Variable p of type pointer is pointing to the address of an integer array arr.

Pointer to a function

Void show (int);
void(*p)(int) = &display; //Pointer p is pointing to the address of a function

Pointer to structure

Struct st{

    int i;
    float f;
  }ref;
  struct st *p=&ref;

Advantage of pointer

1) Pointer **reduces the code** and **improves the performance**, it is used to retrieving strings, trees, etc. and used with arrays, structures, and functions.

2) We can **return multiple values from a function** using the pointer.

3) It makes you able to **access any memory location** in the computer's memory.

Usage of pointer

There are many applications of pointers in c language.

1) Dynamic memory allocation

In c language, we can dynamically allocate memory using malloc() and calloc() functions where the pointer is used.

2) Arrays, Functions, and Structures

Pointers in c language are widely used in arrays, functions, and structures. It reduces the code and improves the performan

NULL Pointer

A pointer that is not assigned any value but NULL is known as the NULL pointer. If you don't have any address to be specified in the pointer at the time of declaration, you can assign NULL value. It will provide a better approach.

```
int *p=NULL;
```
In the most libraries, the value of the pointer is 0 (zero).

How to read the following pointer?

Int (*p)(int (*)[2], int (*)void))

Explanation

This pointer will be read as p is a pointer to such function which accepts the first parameter as the pointer to a one-dimensional array of integers of size two and the second parameter as the pointer to a function which parameter is void and return type is the integer.

Traversing an array by using pointer

```
#include<stdio.h>
void main()
{
int arr[5]={1,2,3,4,5};
int *p=arr;
int i;
printf("printing array elements...\n");
for(i = 0; i< 5; i++)
{
    printf("%d ",*(p+i));
 }
}
```

Pointer to function in C

```
#include<stdio.h>
int addition();
int main()
{
int result;
int (*ptr)();
ptr=&addition;
result=(*ptr)();
printf("The sum is %d",result);
}
int addition()
{
int a, b,c;
printf("Enter two numbers?");
scanf("%d %d",&a,&b);
c=a+b;
return c;
}
```