Pointers to Function Video 1

Abu Bakr Mohamed Ramadan
eng.abubakr@gmail.com

Content:

- **What is function pointer?**
- **Motivation, what is the use cases of function pointer?**
- **Declaration of function pointer in c**
- **Initialization of function pointer in c**
- **Calling a function using the function pointer**
- **Example on function pointer**
- **Function pointer as arguments**

What is function pointer?

- A function pointer is one of the most important pointer tools which is often ignored and misunderstood by the people.
- A function pointer is similar to the other **pointers** but the only difference is that it points to a function instead of the variable.
- Unlike normal pointers, a function pointer points to code, not data. Typically a function pointer stores the start of executable code.

Motivation, what is the use cases of function pointer?

- Interrupt Vector table is an array of pointers to function (ISR),
- Callback mechanism in modular layered application,
- Basic concept in OS and RTOS programing,
- we can achieve OOP concepts like inheritance and polymorphism in C using function pointers.

Declaration of function pointer in c

- int fpData(int);
- int *fpData(int);
- int (*fpData)(int);
- int (*fpData)(char *);
- int* (*fpData)(char *);
- int (*fpData)(int, char *);

Initialization of function pointer in c

- declaration of function pointer

```
int (* pfAddTwoNumber) (int, int);
```

- Initialize the function pointer with the address of AddTwoNumber

```
pfAddTwoNumber = & AddTwoNumber;
```

or

```
pfAddTwoNumber = AddTwoNumber;
```

- We can also initialize the function pointer in a single line.

```
int (* pfAddTwoNumber) (int, int) = AddTwoNumber;
```

Calling a function using the function pointer

- First, write the name of a function pointer.

  pfAddTwoNumber;

- We know that function pointer is also similar to another pointer but the only difference is that function pointer pointed to a function except for the variable. So we put the * as a prefix to the name of function pointer to take the value of the function pointer.

  ***pfAddTwoNumber;**

- Finally, we have a function. For the function calling just simply add the arguments list with extra parenthesis to the above expression.

  (*pfAddTwoNumber)(10, 20);

Example

```
#include <stdio.h>

#include <stdlib.h>

//function used to add two numbers

int AddTwoNumber(int iData1 ,int iData2)

{

return (iData1 + iData2);

}
```

```
int main(int argc, char *argv[])

{

int iRetValue = 0;

//Declaration of function pointer

int (*pfAddTwoNumber)(int,int) = NULL;

//initialise the function pointer

pfAddTwoNumber = AddTwoNumber;

//Calling the function using function pointer

iRetValue = (*pfAddTwoNumber)(10,20);

//display addition of two number

printf("\n\nAddition of two number = %d\n\n",iRetValue);

return 0; }
```

Function pointer as arguments

- we can pass the function pointer as an argument into the function like the other pointers.
- Example on Function pointer as arguments