Pointers to Function
Video 2

Abu Bakr Mohamed Ramadan
eng.abubakr@gmail.com

Content:

- Using an array of pointers to functions,
- Example on array of pointers to functions,
- Using typedef with the function pointer,
- Declaring function pointers in structure,
- Example on using function pointers in structure,

Using an array of pointers to functions,

- //Declaration of array of function pointer

  int (*apfArithmatics [3])(**int**,**int**)

- //Initialization of array of function pointer

  int (*apfArithmatics [3])(**int**,**int**) =
  {AddTwoNumber,SubTwoNumber,MulTwoNumber};

- //Calling the Add function using index of array

  iRetValue = (*apfArithmatics [0])(20,10);

- The array of function pointers offers the facility to access the function using the index of the array.

Using an array of pointers to functions,

- **Example on array of pointers to functions,**

Using typedef with the function pointer

- It is convenient to declare a type definition for function pointers like:
- /* function pointer */

  typedef int (*pfunctPtr)(int, int);
- Before:
  - int (*pfunctPtr)(int, int) = AddTwoNumbers;
  - iRetValue = (*pfunctPtr)(a, b);
- After:
  - pfunctPtr Calculation = AddTwoNumbers;
  - iRetValue = Calculation(a,b);

Using typedef with the function pointer

- // typedef of array of function pointers

  typedef int (*apfArithmatics[3])(int,int);

- Before:
  - **int (*apfArithmatics [3])(int,int) = {AddTwoNumber,SubTwoNumber,MulTwoNumber};**
- After:
  - **apfArithmatics** aArithmaticOperation = **{AddTwoNumber,SubTwoNumber,MulTwoNumber};**

Declare function pointers in structure

- A **structure** is a collection of variables under a single name. These variables can be of different types,
- A **structure** is a convenient way of grouping several pieces of related information together.
- In C language we cannot create a member function in the structure but with the help of pointer to function, we can provide the facility to user to store the address of the function.

Declare function pointers in structure

- First define a function pointer for example:
  - typedef int (*pfoperation)(int a , int b);
- Then define the struct,
  - typedef struct S_sMath
    {
    int result ; // *to store the resut*
    pfoperation operation; // *funtion pointer*
    } sMath;
- Then Declare the struct
  - sMath smath_operation;
    - OR
  - sMath * psSmath_operation = NULL;

- Initialize the function pointer:
  - Smath_operation.Operation = add;
    - OR
  - psSmath_operation->Operation = add;
- Calling function using pointer to function declared in a struct
  - Smath_operation.result = Smath_operation.Operation(5,3);
    - OR
  - psSmath_operation->result = psSmath_operation->Operation(5,3);

Declare function pointers in structure

- **Example on using function pointers in structure,**