Programming in C

Functions
(Passing Parameters)

Contents

- Parameter Passing
- Call by Value
- Call by Reference

Parameter (Arguments)

- There are two types of parameters in C:
  - *Actual parameters* are parameters as they appear in function calls.
  - *Formal parameters* are parameters as they appear in function definition.

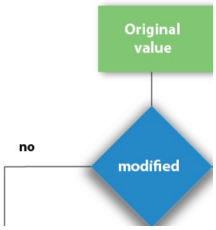
```
void main()
  {
  int n;
  display(n);
  }

  void display(int x)
  {
  ---------
  ---------
  }
```

- **x is "Formal Parameter"**
- **n is "Actual Parameter"**

Types of Parameter Passing

- There are two ways to pass value or data to function in C language: ***call by value* and *call by reference.*** Original value is not modified in call by value but it is modified in call by reference.



Example (call by Value)

```
#include <stdio.h>
void swap(int, int);
void main()
{
int x, y;
printf("Enter the value of x and y\n");
scanf("%d%d",&x,&y);
printf("Before Swapping\nx = %d\ny =
   %d\n", x, y);
swap(x, y);
printf("After Swapping\nx = %d\ny =
   %d\n", x, y);
}
```

```
void swap(int a, int b)
{
int temp;
temp = b;
b = a;
a = temp;
printf("Values of a and b is %d
%d\n",a,b);
}
```

Example (Call By Reference)

```
#include <stdio.h>
void swap(int *x, int *y);
void main ()
{
int a = 100;
int b = 200;
printf("Before swap, value of a :
   %d\n", a );
printf("Before swap, value of b :
   %d\n", b );
swap(&a, &b);
printf("After swap, value of a : %d\n",
   a );
printf("After swap, value of b : %d\n",
   b );
}
```

```
void swap(int *x, int *y)
{ int temp;
temp = *x;
*x = *y;
*y = temp;
}
```

No.	Call by value	Call by reference
1	A copy of value is passed to the function	An address of value is passed to the function
2	Changes made inside the function is not reflected on other functions	Changes made inside the function is reflected outside the function also
3	Actual and formal arguments will be created in different memory location	Actual and formal arguments will be created in same memory location

Assignment

1. Differentiate call by value and call by reference in C with example.

REFERENCES

1. https://www.tutorialspoint.com/cprogramming/c_passing_arrays_to_functions.htm
2. Let Us C , Yashwant Kanetkar
3. https://www.javatpoint.com/call-by-value-and-call-by-reference-in-c
4. https://www.tutorialspoint.com/cprogramming/c_function_call_by_value.htm

- Programming in C, 2011, by J.B. Dixit
- Basics of C Programming, 2011, by J.B. Dixit