# **C** Programming

## **Functions**

## Parameters (Pass by Value and Pass by Reference)

There are two ways to pass parameters in C / C++

- Pass by Value
- Pass by Reference

### 1. Pass by Value

Pass by Value is when a COPY of the parameter is passed to a function. e.g.,

```
/*
Function that passes a parameter using Pass by Value
#include <stdio.h>
// Function signature
void fxn1(int);
int main()
   int num = 0;
   printf("Enter any number\n");
   scanf("%d", & num);
   //Pass a COPY of variable num to the function
   fxn1 (num);
   printf("\nnum contains %d", num);
   return 0;
} // end main()
```

```
// fxn1() changes the contents of the parameter
void fxn1(int n1)
{
    printf("\nn1 contains %d\n", n1);

    //increment n1
    n1++; // n1 = n1 + 1;

    printf("\nn1 contains %d\n", n1);
} // end fxn1()
```

Repl 14.1: https://replit.com/@michaelTUDublin/141-Pass-by-Value#main.c

### 2. Pass by Reference

Pass by Reference is when you pass the ADDRESS of the parameter to the function. e.g.,

```
/*
Function that passes a parameter using Pass by Reference
*/

#include <stdio.h>

// Function signature
// The parameter is telling the compiler that when the function
// is called, the memory address of the parameter will be passed
void fxn1(int *);

int main()
{
   int num = 0;
    printf("Enter any number\n");
    scanf("%d", & num);

//Pass the ADDRESS of variable num to the function
```

```
fxnl(&num);

printf("\nnum contains %d", num);

return 0;

// end main()

// fxnl() uses the address location of num, which is passed to
// this function and accesses its contents using the dereference
// operator
void fxnl(int *nl)
{
   printf("\nnl contains %d\n", *nl);
   //increment nl
   (*nl)++; // nl = nl + l;
   printf("\nnl contains %d\n", *nl);
}
// end fxnl()
```

Repl 14.2: https://replit.com/@michaelTUDublin/142-Pass-by-Reference