**We can pass arguments to the C function in two ways:**

1. **Pass by Value / call by value**
2. **Pass by Reference / call by reference**

**1. Pass by Value**

Parameter passing in this method copies values from actual parameters into formal function parameters. As a result, any changes made inside the functions do not reflect in the caller’s parameters.

// C program to show use

// of call by value

#include <stdio.h>

void swap(int var1, int var2)

{

int temp = var1;

var1 = var2;

var2 = temp;

}

// Driver code

int main()

{

int var1 = 3, var2 = 2;

printf("Before swap Value of var1 and var2 is: %d, %d\n",

var1, var2);

swap(var1, var2);

printf("After swap Value of var1 and var2 is: %d, %d",

var1, var2);

return 0;

}

**Output**

Before swap Value of var1 and var2 is: 3, 2

After swap Value of var1 and var2 is: 3, 2

1. **Pass by Reference**

The caller’s actual parameters and the function’s actual parameters refer to the same locations, so any changes made inside the function are reflected in the caller’s actual parameters.

// C program to show use of

// call by Reference

#include <stdio.h>

void swap(int \*var1, int \*var2)

{

int temp = \*var1;

\*var1 = \*var2;

\*var2 = temp;

}

// Driver code

int main()

{

int var1 = 3, var2 = 2;

printf("Before swap Value of var1 and var2 is: %d, %d\n",

var1, var2);

swap(&var1, &var2);

printf("After swap Value of var1 and var2 is: %d, %d",

var1, var2);

return 0;

}

**Output**

Before swap Value of var1 and var2 is: 3, 2

After swap Value of var1 and var2 is: 2, 3