

## //Call by Value and Call by Reference

/\*

### Call by Value:

**Actual Parameter:** Actual parameter is the value that is passed to the function when it is called. An actual parameter can be a constant, variable and an expression.

**Formal Parameter:** Formal parameter is the variable defined as function parameter during function definition.

In **call by value**, a copy of the value of the actual parameter is passed to the formal parameter. The changes made inside the function through the formal parameter do not affect the actual parameter.

In **call by reference**, address of the actual parameter is passed to the formal parameter. The changes made inside the function through the formal parameter do affect the actual parameter.

\*/

//Function Prototypes

```
void swapByValue(int, int);
void swapByReference(int *, int*);
void mixedCall(int , int*);
```

```
void main(void)
```

```
{
```

```
    int i=8,j=9;
    printf("\nBefore swapByValue, i=%d,j=%d\n",i,j);
    swapByValue(i,j); // Function Calling
    printf("\nAfter swapByValue, i=%d,j=%d\n",i,j);
```

```
    i=8,j=9;
    printf("\nBefore swapByReference, i=%d,j=%d\n",i,j);
    swapByReference(&i, &j); // Function Calling
    printf("\nAfter swapByReference, i=%d,j=%d\n",i,j);
```

```
    i=8,j=9;
    mixedCall(i,&j); // Function Calling
    printf("\nAfter mixedCall, i=%d,j=%d,&j=%u",i,j,&j);
```

```
}
```

//Function Definitions

```
void swapByValue(int a, int b)
```

```
{
```

```
    int temp;
    temp=a,a=b,b=temp; //another usage of comma operator(Comma operator has Left to Right associativity)
    printf("\nInside swapByValue,a=%d,b=%d",a,b);
```

```
}
```

```

void swapByReference(int *a, int *b)
{
    int temp;
    temp=*a;
    *a=*b;
    *b=temp;
    printf("\nInside swapByReference,a=%d,b=%d",*a,*b);
}

void mixedCall(int a,int *b)//a and b are formal parameters
{
    a=a*a;
    *b= *b * *b;
    printf("\nInside mixedCall, a=%d,*b=%d,b=%u",a,*b,b);
}

```

/\*

Output:

Before swapByValue, i=8,j=9

Inside swapByValue,a=9,b=8

After swapByValue, i=8,j=9

Before swapByReference, i=8,j=9

Inside swapByReference,a=9,b=8

After swapByReference, i=9,j=8

Inside mixedCall, a=64,\*b=81,b=2293528

After mixedCall, i=8,j=81,&j=2293528

Press any key to continue.

\*/