

call by value & call by reference:

call by value

↳ Value of actual parameter is copied in formal parameter.

eg: Swapping of 2 numbers }
 a = 10, b = 20 }
 swap = a = 20, b = 10.
 program name

11/10

#include <stdio.h>

↳ Header file

void swapp(int, int); // Fun. declaration.

main()

{ int a, b;

a = 10;

b = 20;

swapp(a, b);

printf("a = %d b = %d", a, b);

void swapp(int a, int b)

// Fun. definition
 int a = 10, b = 20.
 Temp

{ int temp;

temp = a;

a = b;

b = temp;

printf("a = %d b = %d", a, b);

}

o/p:

a = 20

b = 10.

Call by Reference

→ Passing address of actual parameter to formal parameter.

↓
address
↓
For parameter
↓
actual parameter value copied

// & → address
// * → used to get the value for a particular address

Pgm

```
#include <stdio.h>  
void swap (int *, int *); (main fun. address pass)  
// Fun. declare
```

```
main ()
```

```
{  
    int a, b;
```

```
    a = 10; // 1000
```

```
    b = 20; // 1002
```

```
    swap(&a, &b); // swap(1000, 1002) // Fun. call
```

```
    printf("a = %d, b = %d", a, b);
```

```
}  
void swap (int *a, int *b) → int *a = value (address of a)  
                             int *b = value (address of b)
```

```
{ int temp; // 1004
```

```
    temp = *a; // temp = 10
```

```
    *a = *b; // a = 20
```

```
    *b = temp; // b = 10
```

```
    printf("a = %d, b = %d", *a, *b);
```

```
}
```

O/p..

```
a = 20
```

```
b = 10
```