

9) Differentiate call by value and call by reference with an example.

1:47

Voi LTE  
LTE1 79%

13

### Call by Value

In this approach, the values are passed as function argument to the definition of function.

```
1 #include<stdio.h>
2 void fun(int,int);
3 int main()
4 {
5     int A=10,B=20;
6     printf("\nValues before calling %d, %d",A,B);
7     fun(A,B);
8     printf("\nValues after calling %d, %d",A,B);
9     return 0;
10 }
11 void fun(int X,int Y)
12 {
13     X=11;
14     Y=22;
15 }
```

Output

Values before calling 10, 20  
Values after calling 10, 20

Address	48252	24688		
Value	10	20	10	20
Variable	A	B	X	Y

14

### Call by Reference / Address

In this approach, the references / addresses are passed as function argument to the definition of function.

```
1 #include<stdio.h>
2 void fun(int*,int*);
3 int main()
4 {
5     int A=10,B=20;
6     printf("\nValues before calling %d, %d",A,B);
7     fun(&A,&B);
8     printf("\nValues after calling %d, %d",A,B);
9     return 0;
10 }
11 void fun(int *X,int *Y)
12 {
13     *X=11;
14     *Y=22;
15 }
```

Output

Values before calling 10, 20  
Values after calling 11, 22

Address	48252	24688	48252	24688
Value	10	20	11	22
Variable	A	B	*X	*Y

15

Write a program to swap value of two variables using pointer / call by reference.

```
1 int main()
2 {
3     int num1,num2;
4     printf("Enter value of num1 and num2: ");
5     scanf("%d %d",&num1, &num2);
6
7     //displaying numbers before swapping
8     printf("Before Swapping: num1 is: %d, num2 is: %d\n",num1,num2);
9
10    //calling the user defined function swap()
11    swap(&num1,&num2);
12
13    //displaying numbers after swapping
14    printf("After Swapping: num1 is: %d, num2 is: %d\n",num1,num2);
15    return 0;
16 }
```

Output

Enter value of num1 and num2: 5  
10  
Before Swapping: num1 is: 5, num2 is: 10  
After Swapping: num1 is: 10, num2 is: 5

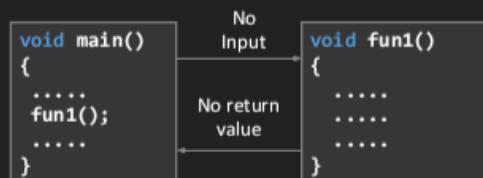
16

### Pointer to Function

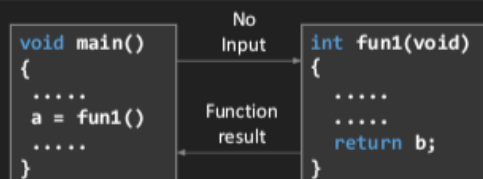
10) Describe different categories of user defined function.

## Category of Function

(1) Function with no argument and but no return value

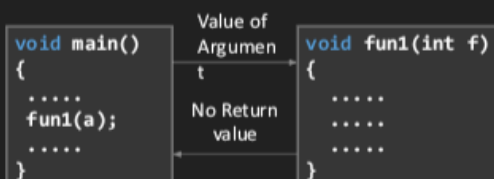


(2) Function with no argument and returns value



## Category of Function cont.

(3) Function with argument and but no return value



(4) Function with argument and returns value

