Session 8

Mostafa Akram

C Functions

Syntax

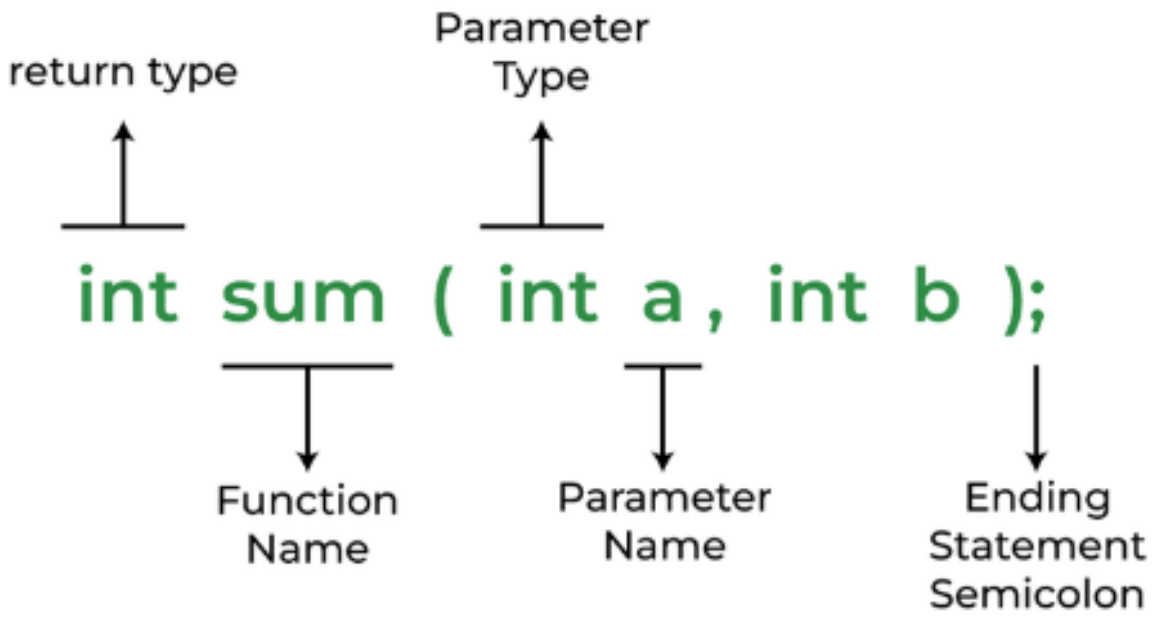
```
return_type name_of_the_function (parameter_1, parameter_2);
```

The parameter name is not mandatory while declaring functions. We can also declare the function without using the name of the data variables.

Example

```
int sum(int a, int b);  // Function declaration with parameter names
int sum(int , int);      // Function declaration without parameter names
```

C Functions

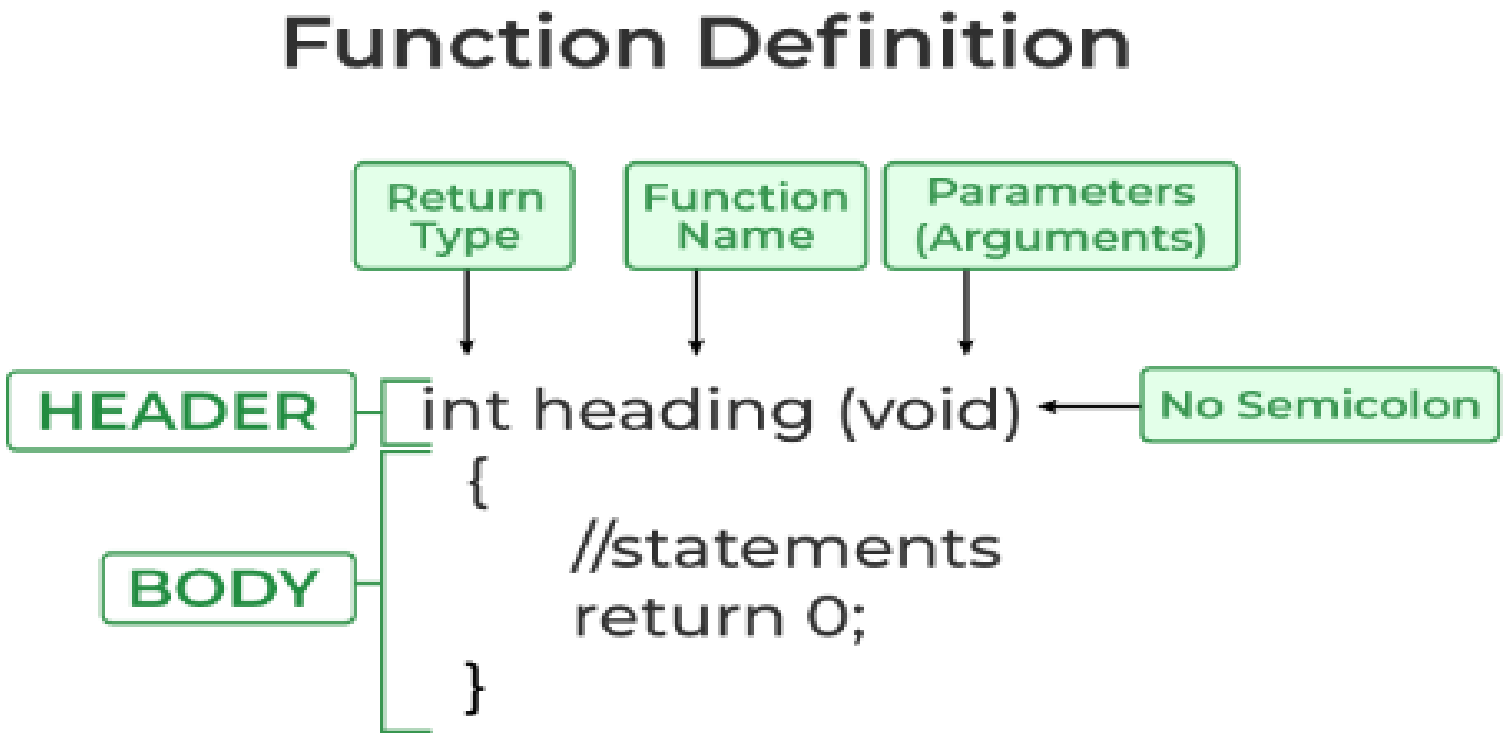


Function Declaration

C Functions

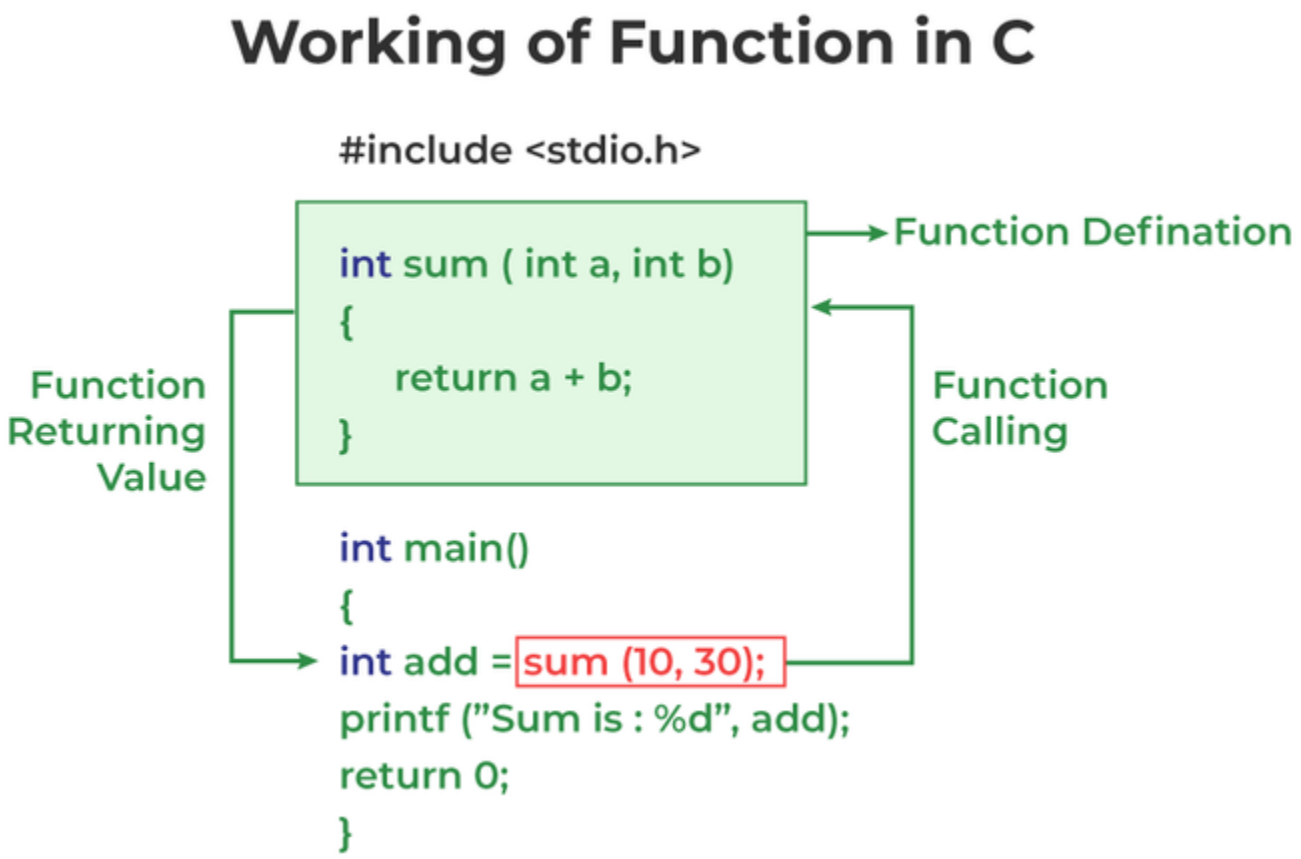
```
return_type function_name (para1_type para1_name, para2_type para2_name)
{
    // body of the function
}
```

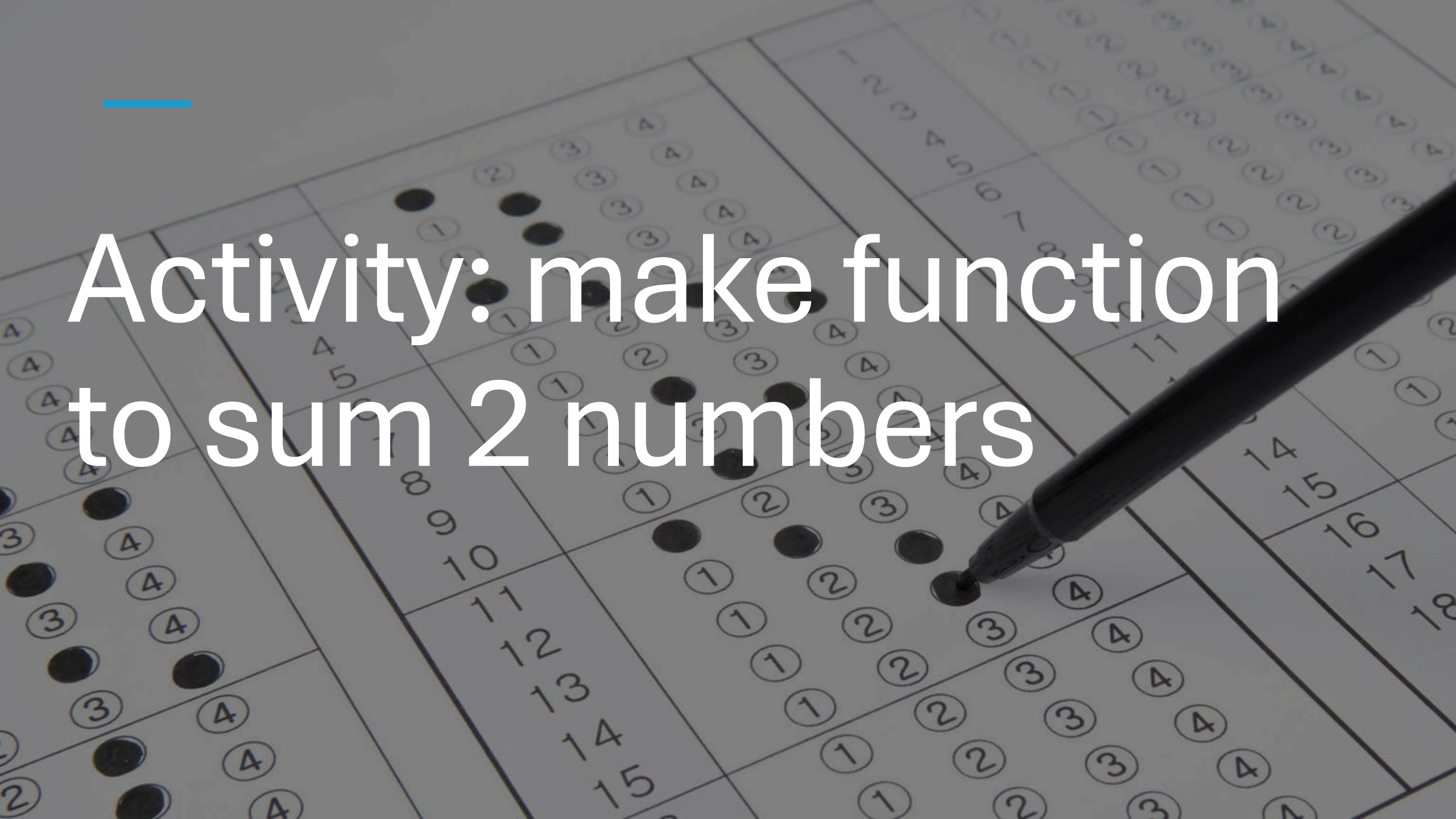
C Functions



Function Definition in C

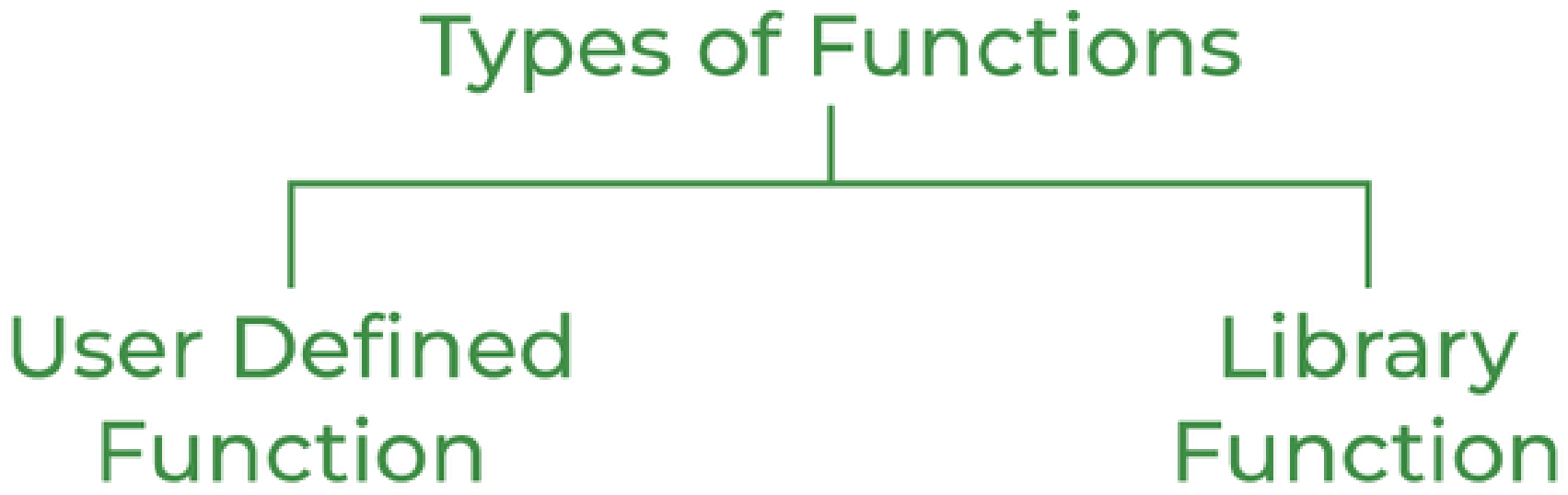
Functions Call





Activity: make function to sum 2 numbers

Types of Functions



1. Library Function

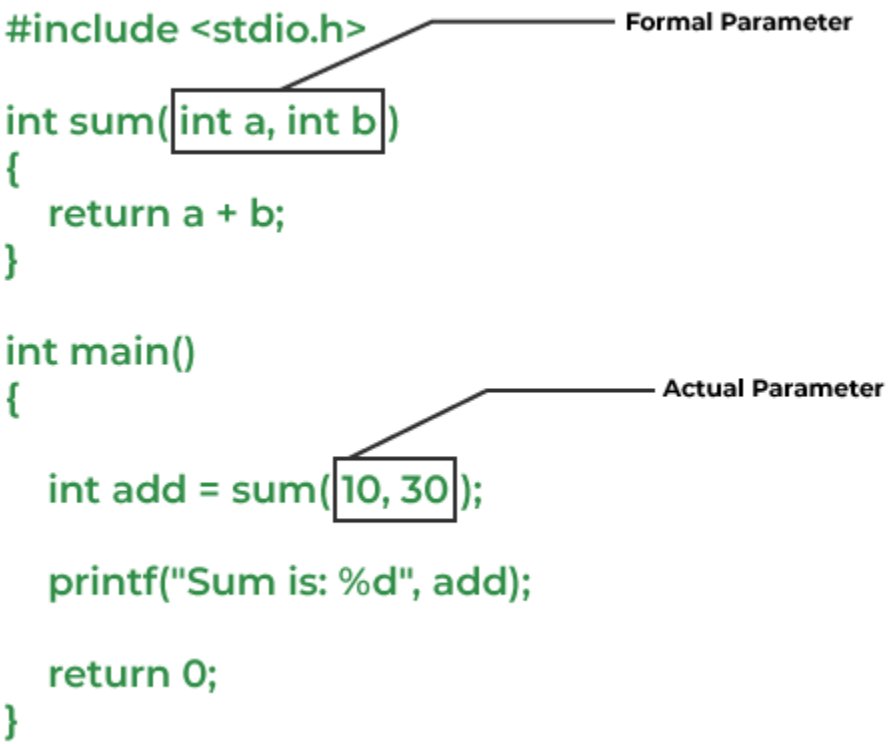
For Example:

```
pow(), sqrt(), strcmp(), strcpy() etc.
```

2. User Defined Function

```
int sum(int a, int b)
{
   return a + b;
}
```

Passing Parameters to Functions



1. Pass by Value

```c
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
```

2. Pass by Reference

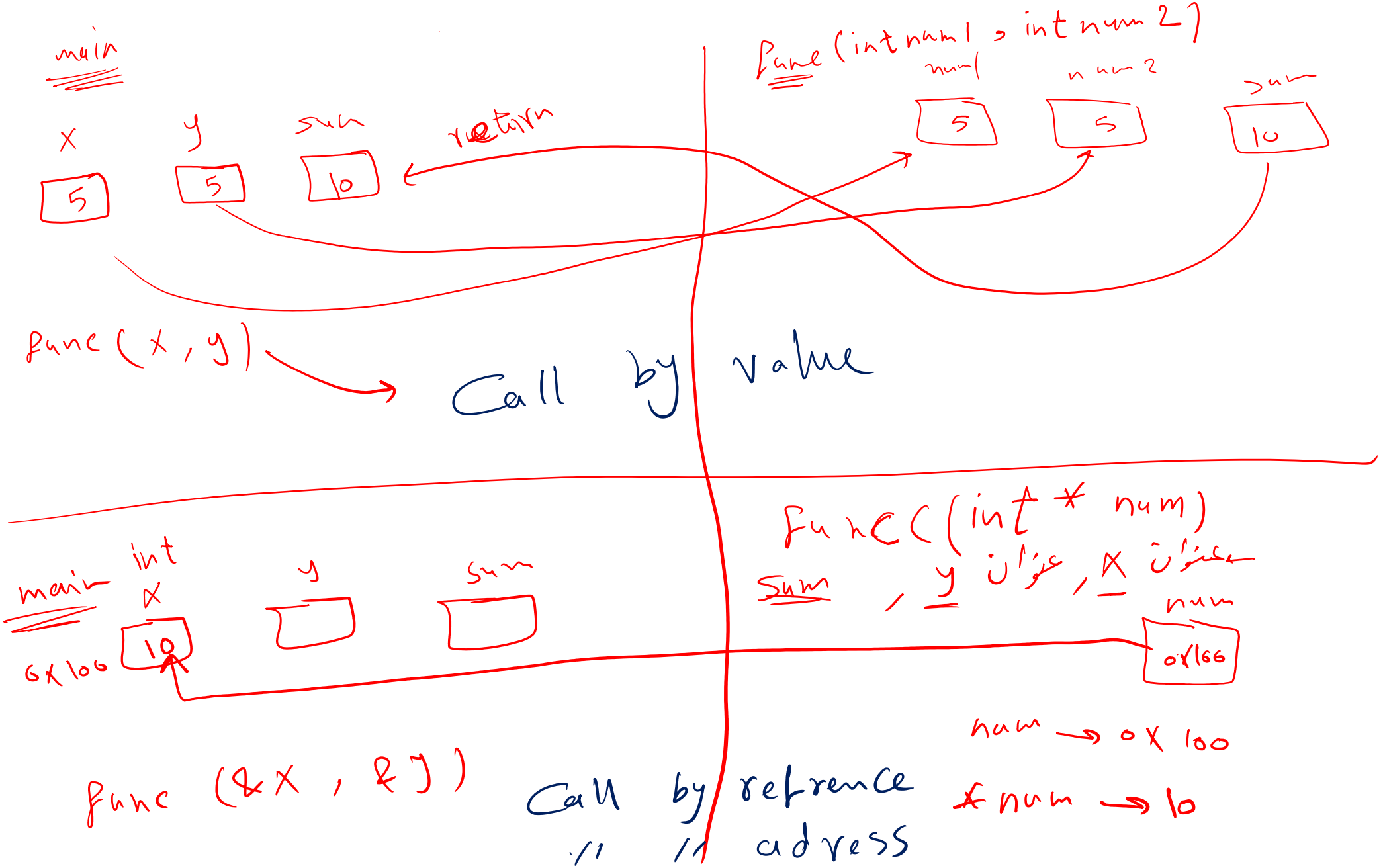
```c
// C program to show use of
// call by Reference
#include <stdio.h>

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
```


main
x
y
sum
5
5
10
return
Func (int num1 , int num2)
num1
num2
sum
5
5
10
func (x , y)
Call by value
main
int x
y
sum
10
0x100
FuncC (int * num)
Sum , y , x
num
0x166
num → 0x100
*num → 10
func (&x , &y)
Call by refrence
// // address

Task

- Let user enter 2 numbers and store them in 2 variables then call function to swap the values of the 2 variables

Links