## How the program works?

In this program we have two files, one is the .c file and the other is .asm file. In the .c file, there is a declaration of a function called add with two long int as the parameters. However, the function is not defined in .c file. In the main function we call the add function whose definition is provided in the .asm file in the assembly language. In .asm file the execution starts from the add label as it has been declared global and according to the calling convention in C the parameters are already passed to the registers rdi and rsi. This function returns the value from the rax register. So, we move the value of rdi register in rax and then add to it the value in rsi register. This way the C program gets the return value and the answer is printed on to the console using the printf function.

So for this program to work we need to link their object files together which is done by the below described gcc commands.

## Commands

**nasm -felf64 main.asm** This command coverts the .asm file to the .o file which is the object code. To see the output, i.e, the .o file type in "**make assemble**".

**gcc prog-add.c main.o** This command is used to link both the object files so that an executable file is created and the program can run on the shell. This command can be called by typing in "**make link**". a.out file will be created after this.

**./a.out** This is the executable file created after both the object files are linked. This command is used to run the file. To run this file type in "**make run**". This will demand two int values as input and will then return the output, i.e, the sum of the two ints.

To clear all the files created in the process type in "**make clear**".