Assembly code explanation

In this code, the execution begins with the _start function; it moves the first integer value '2' to register rdi and the second value integer '5' to register rsi. After that, it calls the function checkGreater. checkGreater function compares the value stored in rdi and rsi. If rdi>rsi, it jumps to the L1 label , else jumps to the L2 label. The L1 label prints the string "1" using the writing system call, whereas the L2 label prints the string "0" using the same system call. After complete execution, control goes back to checkGreater when either one of the labels above is executed after that control is returned to _start and _start exits using the EXIT system call.

C code explanation

In this code, I have first declared the function checkGreater with return type void after that, calling the function in the main() function.

Error explanation

main() cannot call checkGreater as linker throw an error "multiple definitions of _start" because a program cannot have multiple entry points. At the time of compilation, when the C program calls the checkGreater, it not able to call as the assembly program starts with the _start , which in turn tells the kernel to start the code from _start , not from checkGreater.