

Task 1

1. Lines 5-13 define constants and equates that are used throughout the program. The purpose of these lines is to make the code more readable and maintainable by assigning symbolic names to certain values or system calls. Instead of using numeric values directly in the code, these symbolic names make it clear what the code is doing.
2. "resp" is used to allocate a specified number of bytes. In this case, it's reserving 4 bytes of memory and is labeled as "siffer". "Siffer" can hold data that will be read, processed, and displayed by the program.
3. Lines 35-39 work together to print the message to the terminal. Line 35 loads the length of the message into the 'edx' register. Line 36 loads the memory address of the message into the 'ecx' register. Line 37 loads the file descriptor for the standard output into the 'ebx' register. Line 38 loads the system call number for writing to the standard output into the 'eax' register. Line 39 triggers the system call to write the message to the standard output. The system call is executed, and the message is printed to the terminal.
4. After executing the instruction on Line 43, the program counter will jump to Line 86.
5. Lines 86-122 are included in the "lessiffer" block.
6. The ret instruction on Line 122 returns to Line 50.