Sondre Langedal Ness & Jobjørn Røkenes Myren

Task 1

1. Explain Lines 5–13 and what is the purpose of these lines?
   - The lines 5-13 declares constants to be used throughout the program.

2. What does Line 17 do? In particular, explain resb in this line.
   - siffer resb 4 - reserves 4 bytes to the variable named "siffer".

3. Which lines in the program are supposed to print the messages below? Explain your answer." Skriv to ensifrede tall skilt med mellomrom ."
   " Summen av tallene maa vaere mindre enn 10."
   - Line 35-39 are supposed to print the message.
     The address "meldlen" witch contains the length of the message is stored in the register edx. Then the address meld witch contains the message is stored in the register ecx.
     The STDOUT constaint is then moved to the register ebx. This specifies that the output should go to the standard output by moving the value of STDOUT into the ebx register.
     The SYS_WRITE constaint is then moved to the register eax. This prepares the SYS_WRITE system call by moving its value into the eax register.
     Next we call int 80h - This instruction triggers the system call. In this context, it will execute the SYS_WRITE system call, which writes the message to the standard output

4. After executing the instruction on Line 43, which line the program counter will jump to?
   - 86, where we find the lessiffer method.

5. Which lines are included in the lessiffer block?
   - 86-122

6. The block Feil on Line 113 is called conditionally from Lines 104 & 106. Where does the ret instruction on Line 122 return to?
   - when the lessiffer method is being called throughout the program, the next address after the call statement is pushed to the stack. This address is where the ret instruction will return
     With this in mind, lines 104 and 106 determine if the input is a valid single-digit number. If the input doesn't meet this criteria, the program jumps to the "Feil" block at line 113, where an error message is printed.
     After printing the error message, the function encounters the ret instruction at line 122. This ret instruction pops the return address from the stack and transfers control to that address.
     Overall line 122 will return to the instruction immediately following the call lessiffer instruction that was used to invoke the lessiffer function in the first place.