

Oppgave 1

1. Linjene 5-13 definerer noen konstanter. De gir symbolske navn til visse tall og systemkall for å gjøre koden mer lesbar. For eksempel brukes `cr` og `lf` for å representere vognretur (carriage return) og linjeskift (line feed). Videre blir systemkall som `SYS_EXIT`, `SYS_READ`, og `SYS_WRITE` også tildelt symbolske navn. Også standard inngang (`STDIN`), standard utgang (`STDOUT`), og standard feilutgang (`STDERR`) er gitt symbolske navn. Denne delen av koden er nyttig for å gjøre koden mer forståelig ved å erstatte tall med meningsfulle navn.
2. Linjen "siffer resb 4" i programmet tildeler minne i bss (Block Started by Symbol) segmentet. Ordet "resb" står for "reserve bytes," og det brukes til å reservere en bestemt mengde minne for variabelen "siffer." I dette tilfellet reserveres 4 byte (32 bits) med minne til "siffer."

3.

```
mov edx,meldlen
mov ecx,meld
mov ebx,STDOUT
mov eax,SYS_WRITE
int 80h
```

Disse linjene setter opp parametere for systemkallet `SYS_WRITE` for å skrive ut meldingen som er lagret i datasegmentet.

4. På linje 43 vil programtelleren hoppe til "Slutt" etiketten hvis innlesningen på linje 38 ("call lessiffer") ikke lykkes. "Slutt" etiketten er der programmet avslutter ved å returnere fra hovedfunksjonen til systemet.
5. Lessiffer-blokken i koden din starter med etiketten "lessiffer:" og slutter med "ret". Denne blokken inneholder koden for å lese inn et siffer fra tastaturet, sjekke om det er et gyldig tall (0-9), og konvertere det til et tall. Hvis inntastingen ikke er gyldig, blir en feilmelding skrevet til feilmeldingsutgangen (`STDERR`).
6. Returinstruksjonen på linje 122 vil returnere til adressen som kalte funksjonen "skrivsiffer." I dette tilfellet ble funksjonen "skrivsiffer" kalt fra linje 106. Så, returinstruksjonen på linje 122 vil returnere kontrollen til linje 106.