

4 Kirjoita oheinen ohjelma kahdeksi vapaasti sijoiteltavaksi moduuliksi (toiseen pääohjelma, toiseen aliohjelma ja keskeytysten palvelut).

```

$sequit.inc    name    harj_8_3          ; Ohjelma lähettää päätteelle jatkuvana virtana
               ; sitä merkkiä, joka on viimeksi näppäilty
               org     h'20000
               dc.l    alku
               org     h'020150
               bra     ERI_pal
               org     h'020154
               bra     RXI_pal
               org     h'020158
               bra     TXI_pal

               org     h'20a10
alku:          mov.l    #h'fff000, ER7
               bsr     alusta_SCI        ; sarjaportin alustus
               andc    #b'01111111, CCR ; sallitaan keskeytykset
               bset    #6, @SCI_SCR1    ; RIE = 1
               bset    #7, @SCI_SCR1    ; TIE = 1
               bra     $

alusta_SCI:   :
               rts

ERI_pal:     push.w   R0                ; vastaanottimen virhetilan palvelu
               :
               rte

RXI_pal:     mov.b    @SCI_RDR1, R0L     ; luetaan merkki vastaanottimesta
               bclr   #6, @SCI_SSR1    ; kuitataan keskeytyspyyntö (RDRF=0)
               rte

TXI_pal:     mov.b    R0L, @SCI_TDR1    ; merkki lähetysrekisteriin
               bclr   #7, @SCI_SSR1    ; lähetetään merkki (TDRE=0)
               rte
               end

```

5 Mitkä ovat ne käskyrivit, jotka tuottavat konekielistä koodia CSTARTUP-moduulin alla näkyvän osuuden käänöksessä? Merkitse sitten (vaikka tähdellä) ne käskyt, jotka tulevat myös suoritetuiksi, kun ohjelma ajetaan.

```

#define H8_LARGE
#define H8S_MAX
;#define IGNORE_LOW_INIT
;#define IGNORE_SEG_INIT

program CSTARTUP

#ifndef IGNORE_LOW_INIT
#ifndef H8S_MAX
mov.l    # 0x00FFFF40,er7
#endif
#endif
#ifndef H8S_64k
mov.w    # 0xFF40,r7
#endif
extern low_level_init
jsr @low_level_init
#endif

#ifndef H8_LARGE
mov.l    #SFE CSTACK,er7
#else
mov.w    #SFE CSTACK,r7
#endif

#ifndef IGNORE_LOW_INIT
mov.b    r6l,r6l
beg no_seg_init
#endif

#ifndef IGNORE_SEG_INIT
jsr @seg_init
#endif

no_seg_init:
jsr @main
:
endmod

```

```

/*
 * low_level_init makes it possible to customize
 * initialization before main function is executed.
 */

#pragma language=extended
#pragma codeseg(RCODE)

unsigned char far_func low_level_init (void)
{
    /*=====*/
    /* Check your watch dog if needed */
    /*=====*/

    /*=====*/
    /* Init I/O which must be enabled */
    /* before segment initializations */
    /* (as address lines). */
    /*=====*/

    /*=====*/
    /* Choose if segment initialization */
    /* should be done or not. */
    /* Return: 0 to omit seg_init */
    /* 1 to run seg_init */
    /*=====*/
    return (0);
}

```