1.

(a) Whenever there is an interrupt from INT0, the program reads the value from PINA, and store that value to data memory where Y-register is pointing with post-increment, and counting how many times interrupt is occurred with storing the counting result in data memory where X-register is pointing. After these processes, the program returns from ISR.

(b)

i.

```
<1> LDI    mpr, $03 ; 0000 0011
<2> STS    EICRA, mpr
```

ii.

```
<3> LDI    mpr, $01
<4> OUT    EIMSK, mpr
```

iii.

```
<5> LDI    mpr, $00
<6> OUT    DDRA, mpr
```

iv.

```
<7> SEI
```

2.

(a)

DevA: INT0

DevB: INT3

DevC: INT5

(b)

DevB's interrupt is detected on a falling edge.

(c)

Port D and Port E

(d)

<1> OUT DDRE, mpr

<2> OUT PORTD, mpr

(e)

<3> LDI mpr, $29 ;0010 1001, INT5, INT3, INT0

<4> OUT EIFR, mpr

(f)

Because of interruption hierarchy in AVR board, the subroutine ISR_DevA will be executed first.

3.

```
Initialize:
;STACK INITIALIZATION
        LDI     mpr, LOW(RAMEND)
        OUT     SPL, mpr
        LDI     mpr, HIGH(RAMEND)
        OUT     SPH, mpr
;LOAD COUNTER
        LDI     counter, 100
;NORMAL MODE, PRESCALE 1024, OC0 DISCONNECTED
        LDI     mpr, $07    ;0000 0111
        OUT     TCCR0, mpr
; SET TOIE0
        LDI     mpr, $01    ; 0000 0001
        OUT     TIMSK, mpr
;INITIALIZE TCNT0 TO COUNT 10ms
        LDI     mpr, 100
        OUT     TCNT0, mpr
        SEI
LOOP:
        CPI     counter, 0
        BRNE    LOOP
Reload_counter:
        LDI     mpr, 100
        OUT     TCNT0, mpr
        LDI     mpr, $01 ; 0000 0001
        OUT     TIFR, mpr
        DEC     counter
        RETI
```

4.

```
initUSART1:
;STACK INITIALIZATION
        LDI     mpr, LOW(RAMEND)
        OUT     SPL, mpr
        LDI     mpr, HIGH(RAMEND)
        OUT     SPH, mpr
;INITIALIZE I/O DEVICE
        LDI     mpr, (1 << PD3)
        OUT     DDRD, mpr
;SET BAUD RATE
        LDI     mpr, LOW(103)
        STS     UBRR1L, mpr
        LDI     mpr, HIGH(103)
        STS     UBRR1H, mpr
;ASYNCHRONOUS, 2 STOP BIT, 8 BIT DATA, EVEN PARITY
        LDI     mpr, (0 << UMSEL1 | 1 << USBS1 | 1 << UCSZ11 | 1 << UCSZ10 | 1 << UPM11 | 0 << UPM10)
        STS     UCSR1C, mpr
;ENABLE TRANSMITTER MODE & DATA REGISTER EMPTY INTERRUPT
        LDI     mpr, (1 << TXEN1 | 1 << UDRIE1)
        STS     UCSR1B, mpr
        SEI
```