

a)

	LDI R16, 00	R16 = 0
	LDI R17, 20	R17 = 20
L2:	INC R16	R16++
	DEC R17	R17--
	CP R16, R17	R16 - R17
	BRSH L1	if (R16 >= R17) then go to L1
	RJMP L2	go to L2
L1:	ADD R16, R17	R16 = R16 + R17

حلقه تکرار می شود تا این که R16 و R17 برابر 10 شوند ((R16 = R17 و  
BRSH اجرا می شود و از حلقه خراج شده و بعد R16 با R17 جمع می شود و  
مقدار R16 برابر 20 می شود.

b)

LDI R16, 15	R16=\$0F
LDI R17, \$FF	R17=\$FF
CLC	C = 0
loop 1: ROR R17	rotate right
DEC R16	R16--
SBRC R17, 3	skip if (R17(3) == 0)
RJMP loop1	go to loop 1
MOV R17, R16	R17 <= R16

حلقه تکرار می شود تا این که ROR چهارمین بیت R17 را برابر 0 کند و با هر حلقه مقدار R16 کاهش می یابد.

R17: 10111111	R16 = \$0E
R17: 11011111	R16 = \$0D
R17: 11101111	R16 = \$0C
R17: 11110111	R16 = \$0B
R17: 11111011	R16 = \$0A

و بعد از SBRC دستور MOV اجرا می شود و بعد مقدار R17 برابر \$0A می شود.

c)

LDI R27, \$02	R27 = \$02
LDI R26, \$00	R26 = \$00
LDI R16, \$FF	R16 = \$FF
LDI R17, \$0F	R17 = \$0F
STS \$0200, R17	M[0200] <= R17 = \$0F
INC R17	R17 = \$10
ST Z+, R17	M[\$0200] <= \$10 then Z = \$0201
ST Z+, R16	M[\$0201] <= \$FF then Z = \$0202
LD R20, -Z	Z = \$0201 then R20 <= M[\$0201] = \$FF