3A

act 1. Define SP = RAMEND, but the SP already starts at 0x08FF.

act 2. We set the data into stack manually but the stack pointer does not change from $29D when we add so when we pop it pops nothing.

ans

1. When reset SP point to $08FF
2. Push data into stack make SP decrement
3. Pop data make SP increment
4. We can change value of the SP and we might set the SP to some certain value same as activity 2: In activity 2 pop instructions do nothing because position of stack pointer is wrong it does not decrement because we didn’t push

3-B

activity1 make LED switch using PORTB,PORTC,PORTD but we need to put some delays so human eye can see

activity3 read from PIND out to PORTB

1. Upon reset set DDR as input
2. To set input port -> set DDR to 0x00;
3. program to write is just add instruction SBIS PORTB,0 and OUT PORTB,R17($55);

3-c

activity 1 counter 00->255

activity 2 counter 00->10

activity 3 counter with switch 00 -> switch

1. R20 max 255
2. Use portD to display count PORTB for set max count, can use one port for both but we will need to change DDR twice for every number, it will be very inconvenient and unnecessary, so preferably using two port is easier
3. Branch if not equal is not used in my method. I just let the value go to 255 and back to one, and for b I use a branch if equal to 10.

3-D

act 3

1. value in R0 is 0xE6, because the value of multiplication store in R1:R0
2. R0 has 0x0E , R1 has 0x01 because the R1 stores the higher byte of the answer, R0 stores lower byte. The answer should be 30\*9 = 270 which is 10E in hex;