CS150: Computer Organization and Architecture
Exam 2

Name: ___________________________

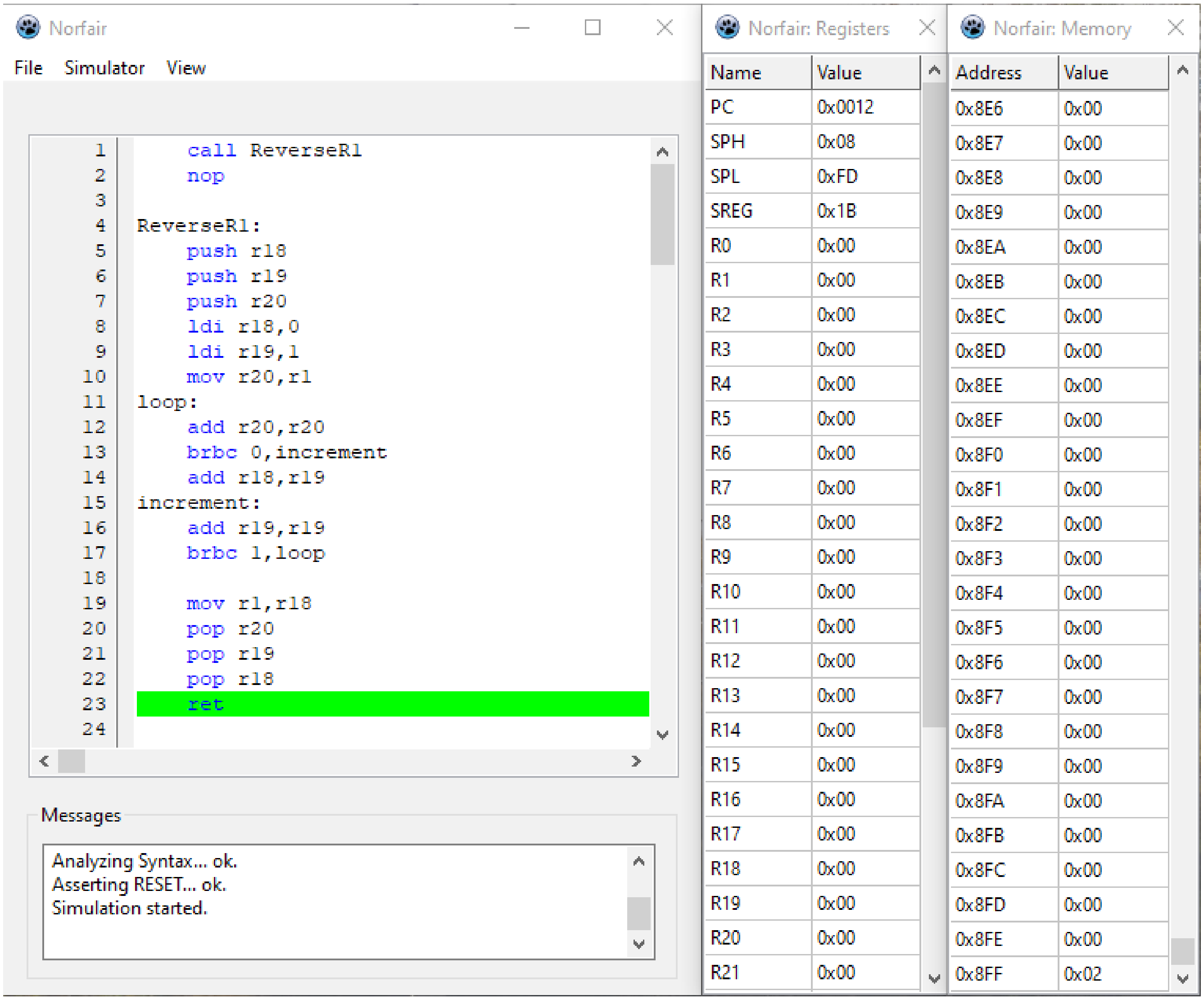
No books, notes, or electronic devices of any kind are to be used. This exam contains 11 questions and is worth 160 points.

1. (5 pts) Write the **machine code** (in binary) for an instruction that will copy the contents of R8 into R15.

2. (5 pts) Write the **machine code** (in binary) for an instruction that will perform a bitwise OR of the value in R17 with the value 0xF4.

3. (10 pts) Write the **machine code** (in binary) for an instruction that will branch to address 0x3B if the result of the last operation was zero. The instruction that you write will be placed at address 0x46 in program memory.

4. (20 pts) Consider the following Norfair screenshot:



```
 1        call ReverseR1
 2        nop
 3
 4    ReverseR1:
 5        push r18
 6        push r19
 7        push r20
 8        ldi r18,0
 9        ldi r19,1
10        mov r20,r1
11    loop:
12        add r20,r20
13        brbc 0,increment
14        add r18,r19
15    increment:
16        add r19,r19
17        brbc 1,loop
18
19        mov r1,r18
20        pop r20
21        pop r19
22        pop r18
23        ret
24
```

```
Analyzing Syntax... ok.
Asserting RESET... ok.
Simulation started.
```

Name	Value
PC	0x0012
SPH	0x08
SPL	0xFD
SREG	0x1B
R0	0x00
R1	0x00
R2	0x00
R3	0x00
R4	0x00
R5	0x00
R6	0x00
R7	0x00
R8	0x00
R9	0x00
R10	0x00
R11	0x00
R12	0x00
R13	0x00
R14	0x00
R15	0x00
R16	0x00
R17	0x00
R18	0x00
R19	0x00
R20	0x00
R21	0x00

Address	Value
0x8E6	0x00
0x8E7	0x00
0x8E8	0x00
0x8E9	0x00
0x8EA	0x00
0x8EB	0x00
0x8EC	0x00
0x8ED	0x00
0x8EE	0x00
0x8EF	0x00
0x8F0	0x00
0x8F1	0x00
0x8F2	0x00
0x8F3	0x00
0x8F4	0x00
0x8F5	0x00
0x8F6	0x00
0x8F7	0x00
0x8F8	0x00
0x8F9	0x00
0x8FA	0x00
0x8FB	0x00
0x8FC	0x00
0x8FD	0x00
0x8FE	0x00
0x8FF	0x02

After the processor fetches and executes the RET instruction that is highlighted on line 23 of the editor, what will be the value of the following four items?

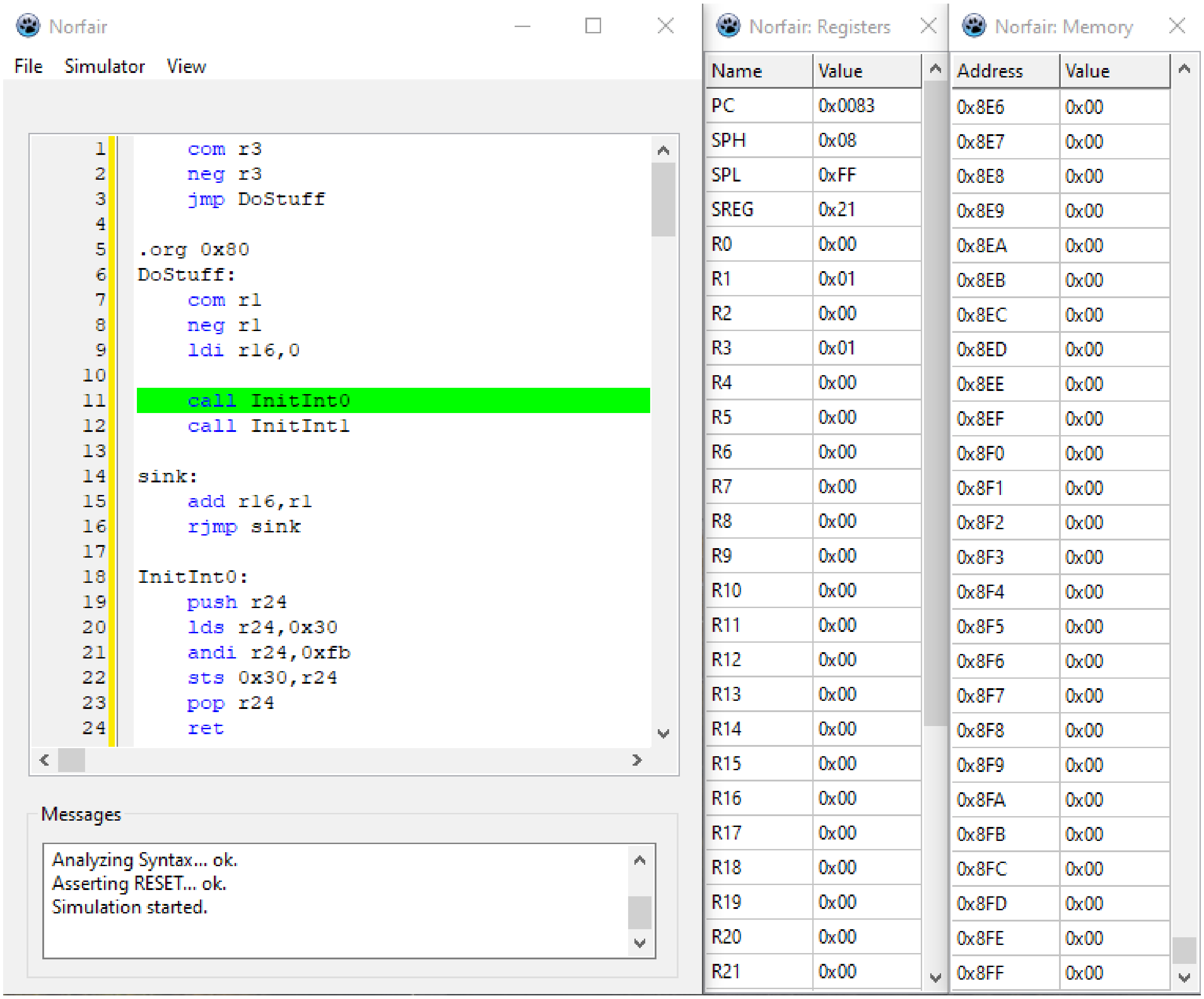
PC: ______________________

SPH: ______________________

SPL: ______________________

R18: ______________________

5. (20 pts) Consider the following Norfair screenshot:



```
 1      com r3
 2      neg r3
 3      jmp DoStuff
 4
 5  .org 0x80
 6  DoStuff:
 7      com r1
 8      neg r1
 9      ldi r16,0
10
11      call InitInt0
12      call InitInt1
13
14  sink:
15      add r16,r1
16      rjmp sink
17
18  InitInt0:
19      push r24
20      lds r24,0x30
21      andi r24,0xfb
22      sts 0x30,r24
23      pop r24
24      ret
```

```
Analyzing Syntax... ok.
Asserting RESET... ok.
Simulation started.
```

Name	Value
PC	0x0083
SPH	0x08
SPL	0xFF
SREG	0x21
R0	0x00
R1	0x01
R2	0x00
R3	0x01
R4	0x00
R5	0x00
R6	0x00
R7	0x00
R8	0x00
R9	0x00
R10	0x00
R11	0x00
R12	0x00
R13	0x00
R14	0x00
R15	0x00
R16	0x00
R17	0x00
R18	0x00
R19	0x00
R20	0x00
R21	0x00

Address	Value
0x8E6	0x00
0x8E7	0x00
0x8E8	0x00
0x8E9	0x00
0x8EA	0x00
0x8EB	0x00
0x8EC	0x00
0x8ED	0x00
0x8EE	0x00
0x8EF	0x00
0x8F0	0x00
0x8F1	0x00
0x8F2	0x00
0x8F3	0x00
0x8F4	0x00
0x8F5	0x00
0x8F6	0x00
0x8F7	0x00
0x8F8	0x00
0x8F9	0x00
0x8FA	0x00
0x8FB	0x00
0x8FC	0x00
0x8FD	0x00
0x8FE	0x00
0x8FF	0x00

After the processor fetches and executes the CALL instruction that is highlighted on line 11 of the editor, what will be the value of the following four items?

PC: ______________________

SPH: ______________________

SPL: ______________________

Memory Location 0x8FF: ______________________

6. (10 pts) What is the Location Counter? Where is it found? What is it used for?

7. (10 pts) Write a numbered list of all phases of the atmega328 instruction processing cycle.

8. (10 pts) Write the **assembly language code** that will clear (set to zero) the upper 3 bits (bits [7:5]) of R0 and leave the remaining bits in R0 (bits [4:0]) unchanged.

9. (10 pts) In assembly language, what do labels represent?

10. (20 pts) Generate the symbol table for the following atmega328 assembly language program.

```
reset_vect:
     rjmp init

.org 0x25
init:
     ldi r16, 8
     eor r15, r16
     com r17
     ori r16, 0xA5
     rjmp f_2

.org 0x3c
f_1:
     add r0, r17
     add r16, r18
     brbc 1, yeet
f_2:
     ldi r16, 6
f_3:
     add r18, r16
     brbc 0, f_3
     in r17, 0x25
     jmp f_1
yeet:
     rjmp yeet
```

11. (40 pts) Write a complete subroutine (or function) in atmega328 assembly language called XorR5andR7 that will compute the exclusive or of the values in R5 and R7 and place the result in R3. Your subroutine **cannot** use the EOR instruction. Your subroutine will effectively compute the value

```
R3 = R5 xor R7
```

without using the EOR instruction. You may only use instructions that are contained in the CS150 AVR instruction subset. You should place your subroutine at address 0x40 in program memory. If you use the EOR instruction for any reason in your subroutine you will not earn any credit for this question.