Richardson Chayden
Mark: 9/10 (total score: 9/10)

+194/1/34+

CS 218 - Worksheet 6

☐0 ☐1 ☐2 ■3 ☐4 ☐5 ☐6 ☐7 ☐8 ☐9
☐0 ☐1 ☐2 ☐3 ☐4 ■5 ☐6 ☐7 ☐8 ☐9
☐0 ☐1 ☐2 ☐3 ☐4 ☐5 ☐6 ☐7 ☐8 ■9
☐0 ☐1 ☐2 ☐3 ☐4 ☐5 ☐6 ■7 ☐8 ☐9
☐0 ☐1 ☐2 ☐3 ☐4 ☐5 ☐6 ■7 ☐8 ☐9

← Encode last 5 of NSHE ID

Enter Name: First Last
Chayden Richardson

For each of the following questions, select only the best answer.

Question 1 Assuming a double-word size, two's compliment representation, what is the decimal representation of 0xFFFFFFFA?

0xFA
1111 1010
0101 flip
1 add 1
0000 0110

- ☐ -3
- ☐ -4
- ☐ -5
- ☒ -6

1/1

Question 2 Assuming IEEE 32-bit floating point format, what is the representation of 0xC0D00000 in decimal? Recall that: n = (1) S 1. F 2 E 127 where S(1-bit) : E(8-bits) : F(23-bits) .

1100 0000 1101 0000 ...
129-127 = 2
neg
-1

- ☐ -5.5
- ☐ -6.5
- ☐ -3.25
- ☐ 1.125

0/1

Question 3 Can a constant be changed during the program execution (yes/no)?

- ☐ Only for self-modifying code.
- ☐ Yes.
- ☒ No.

1/1

Question 4 Which register refers to the top of the stack?

- ☐ rbp
- ☐ rFlag
- ☐ rip
- ☒ rsp

1/1

Question 5 When a macro is invoked, how many times is the code placed in the code segment?

- ☒ Once for each time the macro is used in the code.
- ☐ Once.
- ☐ Depend on specific code in the macro.
- ☐ It is never copied into the code segment.

1/1

Question 6 ♣ Which of the following things happen during **push rax** operation? Check all that apply.

- ☒ The rsp register is decremented by 8.
- ☒ The contents of the rax register are copied to [rsp].
- ☐ The contents of [rsp] is copied to the rax register.
- ☐ The rax register is cleared.
- ☐ The rsp register is incremented by 8.
- ☐ *None of these answers are correct.*

1/1

Question 7 How many **bytes** of data does the pop rax instruction remove from the stack?

- ☐ 2
- ☐ 1
- ☒ 8
- ☐ 16
- ☐ 4

1/1

Question 8 Given the following code fragment, what is in the **eax** and **edx** registers?

```
        list    dd   2, 4, 8, 10, 12

        mov     rsi, 1
        mov     eax, 2
        mov     rcx, 2
lp:     imul    dword [list+rsi*4]
        add     rax, 2
        idiv    dword [list+4]
        inc     rsi
        loop    lp
```

- ☐ eax = 0x00000001 and edx = 0x00000002
- ☐ eax = 0x00000002 and edx = 0x00000004
- ☐ eax = 0x00000002 and edx = 0x00000000
- ☒ eax = 0x00000004 and edx = 0x00000002

1/1

Question 9 Given the previous code fragment, what is in the **rcx** register?

- ☐ rcx = 0xFFFFFFFFFFFFFFFF
- ☐ rcx = 0x0000000000000003
- ☒ rcx = 0x0000000000000000
- ☐ rcx = 0x0000000000000002

1/1

Question 10 Given the previous code fragment, what is in the **rsi** register?

- ☐ rsi = 0xFFFFFFFFFFFFFFFF
- ☒ rsi = 0x0000000000000003
- ☐ rsi = 0x0000000000000002
- ☐ rsi = 0x0000000000000000

1/1