Richardson Chayden
Mark: 10/10 (total score: 5/5)

+90/1/2+

CS 218 - Worksheet 5

0 1 2 **3** 4 5 6 7 8 9 ← Encode last 5 of NSHE ID
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

Enter Name: First Last
Chayden Richardson

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

Question 1 Given the following code fragment, what is in the **ax** register?

```
lst   db   2, 4, 6, 8, 10

mov    al, 2
imul   byte [lst+2]
```

- ☐ ax = 0x0012
- ☐ ax = 0x0004
- ☐ ax = 0x0008
- ☒ ax = 0x000c

1/1

Question 2 Given the following code fragment, what is in the **ax** and **dx** registers?

```
list   dw   12, 14, 16, 18, 20

mov    rbx, list
mov    rsi, 3
mov    ax, word [rbx+rsi*2]
cwd
idiv    word [rbx+2]
```

- ☐ ax = 0x0001 and dx = 0x0002
- ☐ ax = 0x0004 and dx = 0x0001
- ☐ ax = 0x0002 and dx = 0x0000
- ☒ ax = 0x0001 and dx = 0x0004

1/1

Question 3 Given the following code fragment, what is in the **eax** and **edx** registers?

```
list   dd   12, 14, 16, 18, 20

mov    rbx, list
mov    rsi, 2
mov    eax, dword [rbx+rsi*4]
cdq
idiv   dword [rbx+4]
```

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

1/1

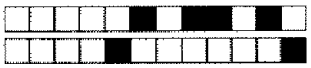
Question 4 Given the following code fragment, what is in the **eax** and **ebx** registers?

```
list   dd   1, 3, 5, 7, 9, 11

       mov    rsi, 4
       mov    eax, 19
       mov    rcx, 2
lp:    sub    eax, dword [list+rsi*4]
       inc    rsi
       dec    rcx
       cmp    rcx, 0
       jne    lp
       mov    ebx, dword [list]
```

- ☐ eax = 0x00000001 and ebx = 0x00000000
- ☐ eax = 0x00000001 and ebx = 0xffffffff
- ☒ eax = 0xffffffff and ebx = 0x00000001
- ☐ eax = 0x-0000001 and ebx = 0x00000001

1/1



Question 5 Given the following code fragment, what is in the **eax** and **ebx** registers?

```
      list   dd   3, 5, 7

      mov    rsi, 0
      mov    eax, 0
      mov    rcx, 3
lp:   add    eax, dword [list+rsi*4]
      inc    rsi
      loop   lp
      mov    ebx, esi
```

- [x] eax = 0x0000000f and ebx = 0x00000003
- [ ] eax = 0x00000003 and ebx = 0x0000000f
- [ ] eax = 0xffffffff and ebx = 0x0000000f
- [ ] eax = 0x00000001 and ebx = 0x00000003

1/1