# Assembly HW #4

1. a)

```
.386
.model flat, stdcall
.stack 4096
ExitProcess PROTO, dwExitCode:DWORD
.code
main PROC
    mov eax, 4
    sub eax, 5
    INVOKE ExitProcess, 0
mainENDP
END main
```

b.)

```
.code
main PROC
    mov al, 0FFh
    add al, 1
    INVOKE ExitProcess, 0
main ENDP
END main
```

c.)

```
.code
main PROC
    mov al, +127
    add al, 1
    INVOKE ExitProcess, 0
mainENDP
END main
```

D

```
mov al, 4
add al, 6
```

4 + 6 = 10$_{10}$

| 0 | 0 | 1 |

count = 2 which is even ∴ PF = 1

2.

```
.code
main PROC
    mov edi, OFFSET MyArray
    mov ecx, LENGTHOF MyArray
    mov ax, 0
L1: add ax, [edi]
    add edi, TYPE MyArray
    loop L1
    invoke ExitProcess, 0
mov WORD, ax
main ENDP
END main
```

3.

| ADDR | DATA |
|---|---|
| myBytes | 0x21 |
| myBytes+1 | 0x67 |
| myBytes+2 | 0xFC |
| myBytes+3 | 0xBA |
| myWords | 7821 |
| +1 | 65 |
| +2 | A9 |
| +3 | 47 |
| +4 | 37 |
| +5 | AC |
| +6 | 5B |
| +7 | 15 |
| +8 | 3C |
| +9 | 0F |

esi = @myBytes

← [esi]

← [esi+1]

lo - 0x67 → AX

hi - 0xFC

AH AL

AX = 8C67

EAX = E1D4A273

myDabes

AX = A273

Ax = 73

Ax = 8629

4, 9, 9, 9, 9, 21, 65, 0CDE, 4, 4, 4

AX = 2

AX = 22

AX = 11

5. EAX = 0000AC22Eh

EDX = 00000AC4h

CX = 022Eh

EAX = FFFFD142h

EDX = 000004242h

CX = FFD1h

(6) AX = 7

AX = 9086h

AX = -51

AX = -22

7.