Muhammad Mufeez

BCS-3J

Coal Lab 05

Q1:

Output:

```
EAX=00001F41  EBX=01105000  ECX=005310AA  EDX=005310AA
ESI=005310AA  EDI=005310AA  EBP=00FEFEE8  ESP=00FEFEDC
EIP=00533672  EFL=00000206  CF=0  SF=0  ZF=0  OF=0  AF=0  PF=1
```

Q2:

Output:

cf = 1, sf = 0, zf = 1, of = 0

```
EAX=00D97F00  EBX=00E67000  ECX=00C910AA  EDX=00C910AA
ESI=00C910AA  EDI=00C910AA  EBP=00D9FCD8  ESP=00D9FCCC
EIP=00C93670  EFL=00000247  CF=1  SF=0  ZF=1  OF=0  AF=0  PF=1
```

cf = 0, sf = 1, zf = 0, of = 1

```
EAX=00D98000  EBX=00E67000  ECX=00C910AA  EDX=00C910AA
ESI=00C910AA  EDI=00C910AA  EBP=00D9FCD8  ESP=00D9FCCC
EIP=00C93678  EFL=00000A92  CF=0  SF=1  ZF=0  OF=1  AF=1  PF=0
```

cf = 0, sf = 1, zf = 0, of = 0

```
EAX=00D98002  EBX=00E67000  ECX=00C910AA  EDX=00C910AA
ESI=00C910AA  EDI=00C910AA  EBP=00D9FCD8  ESP=00D9FCCC
EIP=00C93681  EFL=00000282  CF=0  SF=1  ZF=0  OF=0  AF=0  PF=0
```

Q3:

Output:

COMMENT @

If the array size is word or dword then in word, we will access the next elemen

by [array +2] and so on; for dword [array + 4] and so on

@

Q4:

Output

```
Registers
EAX = 000002F8 EBX = 0000000A ECX = 003F100A EDX = 003F100A ESI = 003F100A EDI = 003F100A EIP = 003F367C ESP = 00B7FEA8 EB

0x003F6015 = 00000000

main.asm
mov eax, arrayD
movzx ebx, arrayW
add eax, ebx
movzx ebx, arrayB
add eax, ebx
mov sum1, eax   ≤ 1ms elapsed
```

```
Registers
EAX = 000005BE EBX = 000000FA ECX = 003F100A EDX = 003F100A ESI = 003F100A EDI = 003F100A EIP = 003F36A

main.asm
add eax, ebx
add eax, [arrayD+4]
mov sum2, eax

mov eax, 0   ≤ 1ms elapsed
mov eax, [arrayD+8]
movzx ebx, [arrayW + 4]
```

```
Registers
EAX = 00000884 EBX = 0000001E ECX = 003F100A EDX = 003F100A ESI = 003F100A EDI = 003F100A EIP = 003F36BF

0x003F601D = 00000000
```

```
main.asm
mov eax, [arrayD+8]
movzx ebx, [arrayW + 4]
add eax, ebx
movzx ebx, [arrayB + 2]
add eax, ebx
mov sum3, eax   ≤ 1ms elapsed

call dumpregs
```

Q7:

Output:

```
Registers
EAX = 00000960 EBX = 01095000 ECX = 00CB100A EDX = 00CB100A ESI = 00000002 EDI = 00CB100A EIP = 00CB367D ESP = 00FAF8DC EBP = 00FAF8E8
```

```
main.asm
mov eax, arrayD[esi * type arrayD]
mov esi, 2
add eax, arrayD[esi * type arrayD]

mov ax, 0   ≤ 1ms elapsed
```

```
Registers
EAX = 000001F4 EBX = 01095000 ECX = 00CB100A EDX = 00CB100A ESI = 00000002 EDI = 00CB100A EIP = 00CB369B ESP = 00FAF8DC EBP = 00FAF8E8 EFL
```

```
main.asm
mov ax, 0
mov esi, 0
mov ax, arrayW[esi * type arrayW]
mov esi, 2
add ax, arrayW[esi * type arrayW]

mov al, 0   ≤ 1ms elapsed
mov esi, 0
```

Registers

```
EAX = 0000018C EBX = 01095000 ECX = 00CB100A EDX = 00CB100A ESI = 00000002 EDI = 00CB100A EIP = 00CB36B5 ESP = 00FAF8DC EBP = 00FAF8E8 EFL = 00
```

main.asm

```
mov al, 0
mov esi, 0
mov al, arrayB[esi * type arrayB]
mov esi, 2
add al, arrayB[esi * type arrayB]
call dumpregs ≤ 1ms elapsed
```