

Q1:

[org 0x100]

jmp start

array\_1 : dw 1,2,3,4,5

array\_3 : dw 0,0,0,0,0

start :

mov ax , [array\_1]

mov bx , [array\_3]

mov dx , 0

mov si , 0

mov bi , 0

mov cx , [array\_1 + si]

l1:

mov ax , [array\_1 + si]

add dx , 3

sub ax , 1

cmp ax , 0

je l2

l2 :

mov bx , [array\_3 + bi]

mov bx , dx

mov dx , 0

add si , 2

add bi , 2

cmp bi , 10

jne l1

mov ax , 0x4c00

int 0x21

Q2:

org 0x100]

jmp start

num1 : dw 101000110101b

start:

mov ax , [num1]

mov cx , 0 ;counting 1s

mov dx , 0 ;counting 0s

mov si , 11

```
l1 :  
ror ax , 1  
jc l2  
add cx , 1  
jmp end
```

```
l2:  
add dx , 1
```

```
end :  
sub si , 1  
cmp si , 0  
jne l1
```

```
mov ax , 0x4c00  
int 0x21
```

Q3:

```
[org 0x100]
```

```
jmp start :  
array_1 : dw 1,2,3,4,5  
size : dw 4  
num_rot : dw 2 ; cant be more than size in any case
```

```
start:  
mov ax , [num_rot]  
mov bi , 0  
mov bx , [array_1 + bi]  
mov cx , 0
```

```
l1:  
mov bx , [array_1 + bi]  
rol bx , 1  
add bi , 2  
add cx , 1  
cmp cx , ax  
jne l1
```

```
mov ax , 0x4c00  
int 0x21
```