Q1. Consider the following program snippet. Assume the offset of "data" label is 0x103. Fill in table of memory contents at the end of this code. All values should be in hex [6 marks]

Group A

```
jmp start

data: dw 0x58A3, 0x7BB1, 0x8E26, 0x9C1A

start:
  AND word [data+2], 0x5555
  XOR byte [data+4], 0xCC
  RCR word [data+6], 2
  SAR byte [data+1], 3
```

Group B

```
jmp start

data: dw 0x7BB1, 0x58A3, 0x9C1A, 0x8E26

start:
  OR word [data+4], 0x8866
  SAL byte [data+1], 3
  XOR byte [data], 0x35
  ROL word [data+6], 2
```

	0	1	2	3	4	5	6	7
DS:103								

RCR = rotate right though carry
SAR = shift right arithmetic
SAL = shift left arithmetic
ROL = rotate left

Group A

	0	1	2	3	4	5	6	7
DS:103	A3	0B	11	51	EA	8E	06	27

Group B

	0	1	2	3	4	5	6	7
DS:103	96	C0	B1	7B	66	8E	6A	70

1 mark for each correct value, max 6

Q2. Consider the following 'thrice' subroutine. It takes in a number as parameter (via stack), and puts 3 times its value in [answer] variable.

[4 marks]

```
1  thrice:
2    push bp
3    mov bp, sp
4
5    ; ** P: store important registers
6    ; ** Q: load input parameter in bx
7    mov ax, bx
8    add ax, bx
9    add ax, bx
10   ; ** R: put output in [answer]
11
12   ; ** S: reload saved registers
13   pop bp
14   ; ** T: return and discard the parameter
15
16 start:
17   ; ** U: pass the value 9 as argument to subroutine
18   call thrice
```

Provide the following missing instruction(s)

Group A
Q, S and T

Group B
P, R and U

Gr A

Q
mov bx, [bp+4]

S
pop bx
pop ax
(in either order)

T
ret 2

1 + 2 + 1 marks

Gr B

P
push ax
push bx
(in either order)

R
mov [answer], ax

U
push 9

2 + 1 + 1 marks