

Q1:

[org 0x0100]

jmp start

data: dw 60, 40, 50, 22, 5

data2: db 3, 4, 1, 9, 6

swapflag: db 0

swap:

cmp dx,1

jne l4

mov ax, [bx+si]

xchg ax, [bx+si+2]

mov [bx+si], ax

jmp l4a

l4:

```
mov al, [bx+si]
xchg al, [bx+si+1]
mov [bx+si], al
```

l4a:

```
ret
```

bubblesort:

```
push ax
push cx
push si
push dx
```

```
dec cx
cmp dx,1
jne mainloop
shl cx, 1
```

mainloop:

mov si, 0

mov byte [swapflag], 0

innerloop:

cmp dx,1

jne l1

mov ax,[bx+si]

cmp ax,[bx+si+2]

jmp l1a

l1:

mov al,[bx+si]

cmp al,[bx+si+1]

l1a:

jbe noswap

call swap

mov byte[swapflag],1

noswap:

    cmp dx,1

    jne l2

    add si, 1

l2:

    add si, 1

    cmp si, cx

    jne innerloop

    cmp byte [swapflag], 0

    je end

    cmp dx,1

    jne l3

    sub cx,1

l3:

    sub cx,1

    jnz mainloop

end:

    pop dx

    pop si

pop cx

pop ax

ret

start:

mov bx, data

mov cx, 5

mov dx,1

call bubblesort

mov bx, data2

mov cx, 5

mov dx,0

call bubblesort

mov ax, 0x4c00

int 0x21

Q2:

[org 0x0100]

jmp start

sum :dw 0

num: dw 7

jmp start

fac:

push bp

mov bp,sp

push ax

push bx

push cx

push dx

mov cx,[bp+4]

sub cx,1

mov ax,0

l1:

cmp cx,0

jne l2

jmp gp

l2:

cmp cx,1

jne fr

jmp gp

gp:

mov [sum],cx

jmp retfac

fr:

mov bx,0

mov dx,1

ll:

addition:

mov ax,dx

;add dx,bx

```
add dx,bx
mov bx,ax
mov [sum],dx
sub cx,1
cmp cx,0
jne addition
retfac:
pop dx
pop cx
pop bx
pop ax
pop bp
ret 2
start:
mov cx,[num]
push cx
call fac

mov ax,0x4c00
```



int 0x21