

Soru 1: SP:SS
BP: SS
BX: DS,ES
DI: DS,ES
SI: DS,ES

Soru 2: MOV AX, [BX]

Soru 3: MOV BX, [BP+4C00H]

Soru 4: 8B7702H

Soru 5: CS registerının değerinin doğrudan değişilmesi sonucu izlenmesi gereken akış değişeceğinden hataya sebep olacaktır.

Soru 6: MOV DI,NUMBER NUMBER değişkeni içerisindeki 16 bitlik değeri DI'ya atıyor.

LEA DI,NUMBER NUMBER değişkeninin offset değerini(adresini) DI'ya atıyor.

Soru 7: MOV AH, ES:[BX]

Soru 8: ADD AX,BX

Soru 9: ADD AL,12H

Soru 10: ADD BP,DI

Soru 11: ADD CX,22H

Soru 12: ADD AL, BYTE PTR [SI]

Soru 13: ADD WORD PTR [DENEME],CX

Soru 14: 8bitlik register(AH) ile 16 bitlik register(CX) ADD AH,CX şeklinde toplanamaz.
Register boyutları uyumsuzdur

Soru 15:

XOR DX,DX

```
ADD AL,CL
ADC DH,0
MOV DL,AL
ADD BL,AH
ADC DH,0
ADD DL,BL
ADC DH,0
```

Soru 16: INC [BX] komutu BX adresindeki verilerin boyut bilgisi olmadığından hatalıdır.

Soru 17: SUB'da(çıkarma işlemi) ilk operandan ikinci operand çıkarılır ve sonuç ilk operandda tutulur, CMP(karşılaştırma işlemi) ise SUB ile aynı işlemi yapar fakat operandlar etkilenebilir, sonuç bayraklarda tutulur.

Soru 18: MULİşaretsiz çarpma işlemidir, IMULİşaretili çarpma işlemidir. Yani operandların işaretli olup olmamasına göre bu komutlardan birini kullanabiliriz. Her iki komut da benzer özellikte olup tek operand almaktadırlar. Akümülatör bu komutların gizli işleneni durumundadır.

Soru 19: DX:AX yazmaçlarıyla ifade edilen 32 bit sayı, 16 bitlik sayıya bölündüğünde sonuç AXbölüm,DXkalan olarak gösterilir . 8 bitlik sayıya bölündüğünde ise sonuç AL bölüm, AHkalan olarak gösterilir.

Soru 20: C = 0, A = 1, S = 0, Z = 0, and O = 0

AX0001 0000 0000 0001 ; DX0010 0000 1111 1111. Word tabanlı işlem yaptığımız için A flag için bytetan taşma(8.bitten 9.bite) durumunu kontrol ediyoruz. Taşma oluşu için A set olacak. İşlem sonucu elde durumu oluşmadığı için C=0 ve toplam sonucu 2byteı geçmediği için de O=0 olacaktır.

Soru 21: MOV AL,DL

```
MUL DL
XOR DH,DH
MUL DX
```

Soru 22:

```
myss SEGMENT PARA STACK 'yigin'
    DW 40 DUP(?)
myss ENDS
myds SEGMENT PARA 'veri'
m    DW 8191
a    DW 884
```

```
c    DW 1
random DW 0
myds ENDS
mycs SEGMENT PARA 'kod'
    ASSUME CS:mycs,DS:myds,SS:myss
MAIN PROC FAR
PUSH DS
XOR AX,AX
PUSH AX
MOV AX,myds
MOV DS,AX
MOV AX,CS:[23]
again: MUL a
ADD AX,c
ADC DX,0
DIV m
MOV AX,DX
CMP AX,256
JA again
MOV random,AX
RETF
MAIN ENDP
mycs ENDS
    END MAIN
```

