**Deney No :** 2

**Deney Tarihi :** 22.10.2015

**Deneyin Amacı :** Döngü komutu loop ve aritmetik komutlar dec,ınc komutlarının kullanımını öğrenmek

**SORU VE CEVAPLAR**

Soru: 1’den 100’e kadar olan sayıların toplamını yapan ve sonucu 0100:1000h bellek adresine yazan Intel x86 programını oluşturunuz.

Cevap:  
  
org 100h

MOV AX, 100h 🡪 ***(VGA Bellek)***

MOV DS, AX 🡪 ***AX Değerini DS değerine atama yapıyorum***

MOV AX,0 🡪 ***AX Değerini 0’a eşitliyorum***

MOV CX, 100 🡪 ***CX Değerini 100’e eşitliyorum. Bu değer loop etiketinin çalışırken kontrol ettiği değeri ifade ediyor***

ETIKET1: 🡪 ***Döngünün döneceği yeri belirtiyorum***

ADD AX,CX 🡪 ***AX Değerine CX değerini ekler ve AX değerine aktarır***

LOOP ETIKET1 🡪 ***Döngü burda CX değerine bakıyor değeri 0 ise döngüyü bitiriyor. Değilse CX değerini 1 azaltarak işlemleri yapıyor***

MOV [1000h], AX 🡪 ***AX’in değerini 1000h adresine atama yapıyorum***

RET  
  
Sorunun Açıklaması : Bu soruda ilk olarak 100h adresinde bir yer açıyoruz. Daha sonra AX yazmacını DS yazmacına eşitleme işlemini yapıyoruz. Daha sonra AX değerini 0’dan, CX değerini 100’den başlatıyoruz. Çünkü 0 ile 100 arasındaki sayıların toplamını bulmak için başlangıçta AX toplam değerimizi tutacak olan yazmacımız oluyor. Daha sonra döngüye başlarken döngüde CX yazmacının değerine bakarak işlem yapılıyor. CX yazmacının değerini 0 ise döngü sonlanıyor eğer değilse 1’er azaltılarak işlemler yapılıyor. Bizim burda döngüde her seferinde AX yani toplam değerimize 100’den geriye doğru başlayarak CX yazmacı 0 olana kadar CX değeri AX değerine eklenerek işlemler yapılıyor ve en sonunda AX değeri yani 0-100 arasındaki sayıların toplamı olan sonucumuz 1000h adresine aktarılıyor.

Soru: 0’dan 5’ekadar olan sayıların faktöriyellerin toplamının 10’a bölümünden kalan sayıyı bulan bir program yazınız.

Cevap:

org 100h

MOV AX,1 🡪 ***AX Değerini 1’e eşitliyorum Çünkü faktöriyel işlemi için 0 dan başlarsa değer her zaman 0 olur***

MOV CX,5 🡪 ***CX Değerini 5’e eşitliyorum***

MOV BX,CX 🡪 ***BX Değerini CX değerine eşitliyorum***

DONGU: 🡪 ***Döngünün döneceği yeri belirtiyorum***

MUL CL 🡪 ***AL ve CL değerlerinin çarpımlarını AX değerine eşitleme yapıyorum***

LOOP DONGU 🡪 ***Döngüde CX Değerini kontrol ediyoruz 0’a eşitse döngüyü bitiyor, değilse 1’er azaltarak işlemleri yapıyor***

ADD DX,AX 🡪 ***DX değerine AX değerini ekler ve DX değerine aktarıyor***

MOV CX,BX 🡪 ***BX değerini CX değerine eşitliyorum***

DEC BX 🡪 ***BX’in değerini 1 azaltıyoruz***

MOV AX,1 🡪 ***AX’in değerini 1 yapıyoruz***

LOOP DONGU 🡪 ***CX değerine bakarak 0 ise döngüyü bitiriyor, değilse 1 azaltarak işlemleri yapıyor***

MOV AX,DX 🡪 ***DX değerini AX değerine eşitliyorum***

MOV BL,10 🡪 ***BL’nin değerini 10’a eşitliyorum***

DIV BL 🡪 ***Burda AH değerine Kalanı AL değerine ise bölümü aktarıyor***

RET

Sorunun Açıklaması : Bu soruda ilk olarak AX değerini 1 den başlatıyoruz AX değeri bizim toplam değişkenimiz oluyor. AX değerini 1 den başlatmamın sebebi döngüde 0 değerinin faktöriyelini hesaplayamadığımızdan 1’den başlattım. Daha sonra CX değerini 5 ten başlatıyorum. CX değeri ise döngüde 1-5 arasındaki sayıların faktöriyellerini tutacak olan değer. Daha sonra her seferinde CX değerini BX yazmacındaki değere eşitliyorum. Bunu yapmamın sebebi ilk olarak CX değerinin faktöriyelini buluyor. Daha sonra 4-3-2-1 hepsinin faktöriyellerini hesaplatmak için BX yazmacını kullanıyorum. Burda BX yazmacı benim faktöriyelini alacağım sayıyı tutuyor. MUL komutu ile çarpma işlemlerini yapıyorum ve AL değerine aktarıyorum. DEC BX komutuyla her seferinde faktöriyeli bulunacak olan sayıyı tutuyorum. En son işlemde DIV BL komutuyla bulduğum faktöriyellerin sonucu olan değeri BL değerine bölerek AH(Kalan) AL(Bölüm) yazmaçlarına aktarma işlemlerini yapıyorum ve sorunun cevabını çıkarıyoruz.