**Derleme komutları**

**python3 ogrenci\_numarası.py**  #terminalden girdileri alarak(varsa) sonucu gösterir.

**python3 ogrenci\_numarası.py<input1.txt** #input1.txt den girdileri alarak terminalde sonucu gösterir.

**python3 ogrenci\_numarası.py<input1.txt>myoutput.txt** """input1.txt den girdileri alarak myoutput.txt dokümanı oluşturarak sonucu buraya yazar """

**diff output1.txt myoutput.txt** #output1.txt dokümanı ile myoutput.txt dokümanını karşılaştırır.

**diff --ignore-all-space output1.txt myoutput.txt** """output1.txt dokümanı ile myoutput.txt dokümanını boşlukları dikkate almadan karşılaştırır. """

**LAB 2 SORU İÇERMEKTEDİR.**

**SORU 1**

Kullanıcıdan alınan veriyi şifreleyen fonksiyonun python kodunu yazınız.

Fonksiyonumuzun adı -> my\_cipher(veri)

Bu fonksiyon yalnızca alfa karakterleri aşağıdaki kurallara göre kodlayacaktır, harfler dışındaki karakterler aynen kalacaktır:

• Tüm alfa karakterler büyük harf olarak değerlendirilecektir.

• İlk alfa karakteri değişmeyecektir (büyük harfe geçiş dışında).

• Sonraki tüm alfa karakterleri, önceki alfa karakterinin alfabetik konumuna göre "Z"ye doğru kaydırılacaktır ("Z" geçilirse "A"ya geri sarılır).

**Örnek:**

he1lo şu şekilde kodlanır:

my\_cipher("he1lo") 🡺 "HM1QA"

h -> H (İlk karakter büyük harfe dönüştürülecek)

e -> M (H alfabedeki bir önceki alfa karakteri ve 8. harftir. E + 8 = M)

1 -> 1 (Harfler dışındaki tüm karakterler olduğu gibi kalsın)

l -> Q (E alfabedeki bir önceki alfa karakteri ve 5. harftir. L + 5 = Q)

o -> A (L alfabedeki bir önceki alfa karakteri ve 12. harftir. O + 12 = A)

**NOT:** isalpha() fonksiyonu verilen karakterin alfabede bulunan bir harf olup olmadığını kontrol eder.

Örneğin ; 'a'.isalpha() 🡺 a bir karakter olduğu için True dönecektir.

**SORU 2**

Kullanıcıdan alınan int sayıyı binary sayıya dönüştüren fonksiyonun python kodunu yazınız.

Fonksiyonumuzun adı 🡺 int\_to\_binary (sayi)

**Örnek:** 13 🡺 1 1 0 1