

ANKARA ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ
BLM1001-101
QUIZ4-GRUP1

Programınızın Ubuntu ortamında çalıştığından emin olunuz. Farklı bir işletim sistemi üzerinde çalışan fakat Ubuntu’da hata ya da uyarı üreten programlar olabileceğini unutmayınız. Bu tip hatalardan dolayı oluşan değerlendirmelere yapılan itirazlar kabul edilmeyecektir.

Doğru çıktı formatı için size verilen örnek girdi ve çıktı dosyalarını dikkatle inceleyiniz. Programınızın doğruluğunu kontrol etmek için aşağıdaki işlemleri gerçekleştirmeniz gerekmektedir.

1) *python3.8 Q1.py < input1.txt*

Bu komut Q1.py dosyasında yazmış olduğunuz programa girdi olarak input1.txt dosyasındaki değerleri almanızı sağlar.

2) *python3.8 Q1.py > myOutput1.txt*

Bu komut programınızın çıktısını myOutput1.txt dosyasına kaydeder.

3) *python3.8 Q1.py < input1.txt > myOutput1.txt*

Bu komut programınıza girdi olarak input1.txt dosyasındaki değerleri alıp, çıktısını myOutput1.txt dosyasına kaydeder.

4) *diff myOutput1.txt output1.txt*

Bu komutu kullanarak kendi çıktınız ile olması gereken çıktıyı karşılaştırınız. Bu komutu girdikten sonra ekranda bir uyarı çıkmıyorsa, programınız bu değerler için doğru çalışıyor demektir. Eğer komutu girdikten sonra komut sisteminde uyarı görüyorsanız bu çıktınızda problem olduğunuz gösterir, kodunuzu düzeltmeniz gerekmektedir.

5) *diff --ignore-all-space myOutput1.txt output1.txt*

Bu komut dosyalar içerisindeki boşlukları (*whitespaces*) dikkate almamayı sağlar. Böylece bir dosyada boşluk var iken diğer dosyada boşluk olmasa bile bu farklılıkları yok sayar.

Kendi oluşturacağınız farklı girdiler için de programınızı test ediniz. Size verilen girdi dosyaları ile değerlendirme sırasında kullanılan girdi dosyaları farklılık gösterecektir.

SORU:

Kullanıcıdan tamsayılardan oluşan bir liste alan ve bu liste içerisinde sağındaki ve solundaki elemanlardan daha büyük olan elemanların oluşturduğu alt listeyi, elde edilen bu alt listenin tekrar eden sayısı varsa tek kalacak şekilde silinmiş halini ve son olarak bu alt listenin büyükten küçüğe doğru sıralanmış halini ekrana yazdıran bir Python programı yazınız.

Örneğin:

add_element() fonksiyonu ile liste doldurulsun

maxfind_subset() fonksiyonu liste içerisinde sağındaki ve solundaki elemanlardan daha büyük olan elemanların oluşturduğu alt listeyi döndürsün

delete_element() fonksiyonu istenen değeri listeden silsin

eliminate_duplicates() fonksiyonu bulduğunuz alt listede tekrarlayan elemanlar varsa **delete_element()** fonksiyonu çağırırsın ve tekrarını silsin, tekrarlamıyorsa bir şey yapmasın.

sort_element() fonksiyonu en son ki listeyi büyükten küçüğe doğru sıralasın

Örnek Input:

[0,9,1,7,3,5,7,12,0,22,9,33,8,22,10]

Output:

[9,7,12,22,33,22]

[9,7,12,22,33]

[33,22,12,9,7]

Not: Eğer **maxfind_subset()** fonksiyonu liste içerisinde sağından ve solundan büyük herhangi bir eleman bulamazsa boş liste dönsün.

maxfind_subset([14,29,30,38,78,104,120,180]) -> []

Not: Kodunuzun yukarıda bahsedilen fonksiyonlar ve liste ile yazılması zorunludur. Bu şekilde yazılmayan kodlar doğru olsa bile **KESİNLİKLE** kabul edilmeyecektir.