Verilen karakter dizisi üzerinde aşağıdaki fonksiyonları tanımlamanızı istiyoruz.

1) islem(kelime, operasyon, indeks) : Diğer bütün fonksiyonların çağırılacağı temel fonksiyonumuzdur, operasyon parametresinin alacağı değerler sadece '\*' ve '+' olabilir. islem fonksiyonu operasyon olarak '\*' değişkeni ile çağırılırsa kelime ve indeks değerlerini kullanarak yildiz fonksiyonunu, '+' değişkeni ile çağırılırsa kelime ve indeks değerlerini kullanarak arti fonksiyonunu çağırmalıdır.

2) yildiz(kelime, indeks): kelime değişkeni ile verilen karakter dizisi üzerinde aşağıdaki işlemleri yapmalıdır:

A) Verilen indeksten sonra kelime içerisindeki '\*' karakteri sayısı en fazla bir ise, olan '\*' karakterini silerek geriye kalan karakterleri karakter dizisi (string) olarak dönmelidir.

Örnek:

yildiz("abcd\*ef", 3)

Çıktı: "abcdef"

Açıklama: 'd', indeksi 3 olan karakterdir ve 'd' karakterinden sonra tek '\*' karakteri olduğu için bu '\*' karakteri silinerek sonuç dönülmüştür.

yildiz("ab\*cd\*efg", 3)

Çıktı: "ab\*cdefg"

Açıklama: 'c', indeksi 3 olan karakterdir ve 'c' karakterinden sonra tek '\*' karakteri olduğu için bu '\*' karakteri silinerek sonuç dönülmüştür. Karakter dizisi içinde toplam 2 tane '\*' karakteri olması önemli değildir.

yildiz("abcdef", 3)

Çıktı: "abcdef"

Açıklama: 'c', indeksi 3 olan karakterdir ve 'c' karakterinden sonra karakter dizisinde '\*' karakteri olmadığı için fonksiyon verilen karakter dizisini aynen dönmüştür.

B) Verilen indeksten sonra kelime içerisindeki '\*' karakteri sayısı en az iki ise, bu indeksten sonraki ilk iki '\*' karakterleri arasında kalan kısmı silip (bu '\*' karakteri de dahil), geriye kalan karakterleri karakter dizisi (string) olarak dönmelidir.

Örnek:

yildiz("abcd\*ef\*hij\*", 3)

Çıktı: "abcdhij\*"

Açıklama: 'd', indeksi 3 olan karakterdir ve 'd' karakterinden sonra 3 tane '\*' karakteri bulunmaktadır, bu nedenle ilk iki yıldız arasında kalan karakterler 'e' ve 'f' ile bu '\*' karakterleri silinerek sonuç dönülmüştür.

yildiz("ab\*cd\*", 1)

Çıktı: "ab"

Açıklama: 'b', indeksi 1 olan karakterdir ve 'b' karakterinden sonra 2 tane '\*' karakteri olduğu için '\*' karakterleri arasında kalan karakterler 'c' ve 'd', bu '\*' karakterleriyle birlikte silinerek sonuç dönülmüştür.

C) Eğer karakter dizisinde indeks ile ulaşılan karakter '\*' ise 1. ve 2. adımdaki işlemler bir sonraki karakter baz alınarak yapılmalıdır.

Örnek:

yildiz("abcd\*ef\*hij\*", 4)

Çıktı: "abcd\*ef"

Açıklama: '\*', indeksi 4 olan karakterdir, o yüzden indeksi 5 olan karakter 'e' üzerinden işlem yapılmalıdır, 'e' karakterinden sonra toplam 2 tane yıldız olduğu için '\*hij\*' kısmı silinerek sonuç dönülmüştür.

yildiz("abcd\*ef\*hij", 4)

Çıktı: "abcd\*efhij"

Açıklama: '\*', indeksi 4 olan karakterdir, o yüzden indeksi 5 olan karakter 'e' üzerinden işlem yapılmalıdır, 'e' karakterinden sonra gelen '\*' karakteri sayısı 1 olduğu için, bu '\*' karakteri silinerek sonuç dönülmüştür.

3) arti(kelime, indeks): kelime değişkeni ile verilen karakter dizisi üzerinde aşağıdaki işlemleri yapmalıdır:

A) Verilen indeksten sonra kelime içerisindeki '+' karakteri sayısı en fazla bir ise, olan '+' karakterini silerek geriye kalan karakterleri karakter dizisi (string) olarak dönmelidir.

Örnek:

arti("abcd+ef", 3)

Çıktı: "abcdef"

Açıklama: 'd', indeksi 3 olan karakterdir ve 'd' karakterinden sonra tek '+' karakteri olduğu için '+' karakteri silinerek sonuç dönülmüştür.

arti("ab+cd+efg", 3)

Çıktı: "ab+cdefg"

Açıklama: 'c', indeksi 3 olan karakterdir ve 'c' karakterinden sonra tek '+' karakteri olduğu için bu '+' karakteri silinerek sonuç dönülmüştür. Karakter dizisi içinde toplam 2 tane '+' karakteri olması önemli değildir.

arti("abcdef", 3)

Çıktı: "abcdef"

Açıklama: 'c', indeksi 3 olan karakterdir ve 'c' karakterinden sonra karakter dizisinde '+' karakteri olmadığı için fonksiyon verilen karakter dizisini aynen dönmüştür.

B) Verilen indeksten sonra kelime içerisindeki '+' karakteri sayısı en az iki ise, bu indeksten sonraki ilk iki '+' karakteri arasında kalan kısım silinip (bu iki '+' karakteri de dahil), yine bu '+' karakterleri arasındaki kısım kalan karakter dizisine eklenerek, elde edilen karakter dizisini (string) dönmelidir.

Örnek:

arti("abcd+ef+hij+", 3)

Çıktı: "abcdhij+ef"

Açıklama: 'd', indeksi 3 olan karakterdir ve 'd' karakterinden sonra 3 tane '+' karakteri bulunmaktadır, bu nedenle ilk iki '+' karakteri arasında kalan karakterler 'e' ve 'f' bu '+' karakterleriyle birlikte silinerek, geriye kalan kısmın (abcdhij+) sonuna 'ef' karakter dizi eklenmiştir ve sonuç dönülmüştür.

arti("ab+cd+", 1)

Çıktı: "abcd"

Açıklama: 'b', indeksi 1 olan karakterdir ve 'b' karakterinden sonra 2 tane '+' karakteri olduğu için artılar arasında kalan karakterler 'c' ve 'd', bu '+' karakterleriyle birlikte silinerek, geriye kalan kısmın (ab) sonuna 'cd' karakter dizisi eklenmiştir ve sonuç dönülmüştür.

C) Eğer karakter dizisinde indeks ile ulaşılan karakter '+' ise 1. ve 2. adımdaki işlemler bir sonraki karakter baz alınarak yapılmalıdır.

Örnek:

arti("abcd+ef+hij+", 4)

Çıktı: "abcd+efhij"

Açıklama: '+', indeksi 4 olan karakterdir, o yüzden indeksi 5 olan karakter 'e' üzerinden işlem yapılmalıdır, 'e' karakterinden sonra toplam 2 tane '+' karakteri olduğu için "+hij+" kısmı silinerek, "hij" karakter dizisi geriye kalan kısmın (abcd+ef) üzerine eklenmiştir ve sonuç dönülmüştür.

arti("abcd+ef+hij", 4)

Çıktı: "abcd+efhij"

Açıklama: '+', indeksi 4 olan karakterdir, o yüzden indeksi 5 olan karakter 'e' üzerinden işlem yapılmalıdır, 'e' karakterinden sonra '+' sayısı 1 olduğu için, bu '+' karakteri silinerek sonuç dönülmüştür.

Açıklamalar:

Fonskiyonlarınız işlemlerini bitirdiği zaman ekrana 'print' etmek yerine, çıktıyı 'return' etmelidir. Kodunuzu yazdıktan sonra test etmek için aşağıdaki örnekte olduğu gibi çağırabilirsiniz, lütfen bütün durumları düşünerek ödevinizi hazırlayınız.

print(islem("abcdef\*", '\*', 3))

Ayrıca forumdaki 2. ödev başlığının altında ilginç girdi/çıktı örnekleri paylaşmanız yararınıza olacaktır. Ödevinizin ekrana herhangi bir açıklama yazdırmasına gerek yoktur, açıklama kısımları tamamen sizlere yardımcı olması için paylaşılmıştır.

Kopya konusuna hiçbir şekilde izin vermiyoruz, otomatik araçlarla bütün kodları tarama yeteneğine sahibiz, o yüzden ödevle kendinizin uğraşıp, sorularınızı forum üzerinden arkadaşlarınızla tartışmanız ya da bize mesaj olarak atmanız sizin faydanıza olacaktır (lütfen soru sorarken dahi kodunuzu paylaşmayınız). Ödevden kopya çektiği tespit edilen kişilerin notu 0 olacaktır. Lütfen birbirinizden veya internetten sadece fikir araştırması (paylaşması) yapınız, arkadaşlarınızdan ya da internetten alacağınız en ufak bir kodu kendi kodunuza eklemeniz ödevinizin benzerlik yüzdesini artıracaktır.

Yüklediğiniz dosyalar '.py' uzantılı Python dosyaları olmalıdır, dosya isim formatı s12345\_isim\_soyisim.py olacak şekilde 12345 yerine öğrenci numaranız, sonrasında da '\_' karakterleri ile ayrılmış şekilde isim ve soyisminizi içermelidir.

Not: Verilen karakter dizilerini isterseniz normal dizilere çevirebilir, karakter dizileri ve diziler üzerinde tanımlanmış count ve index fonksiyonlarını kullanabilirsiniz. Ödevi yaparken 6. haftada işlediğimiz dizileri dilimleme bilgilerine ihtiyaç duyabilirsiniz, ve ayrıca Python listeleri üzerinde ekleme, çıkarma ve iki listeyi birleştirme kısımlarını gözden geçirebilirsiniz. Ancak lütfen fo Bu ödevde döngü kullanarak sizden 4 (dört) farklı görevi yapmanızı istiyoruz. Bunun için programınız önce kullanıcıdan pozitif bir sayı (int n) istemelidir.

Daha sonra kullanıcıdan, ilk girilen sayı kadar daha pozitif sayı istemelidir:

Ödevde, birinci yapmanız gereken, her girilen sayıdan sonra (int n hariç) aşağıda örnekte gösterildiği gibi yıldız basmaktır.

İkinci ve üçüncü göreviniz, girilen sayıların (int n hariç) en küçük ve büyüğünü sırasıyla aşağıda gösterildiği biçimde konsola yazdırmaktır.

Dördüncü ve son göreviniz ise girilen tüm sayıların (int n hariç) toplamını yine aşağıda gösterildiği biçimde konsola basmaktır.

1. görev : 30 puan

2. görev : 25 puan

3. görev : 25 puan

4. görev : 20 puan

Örnek Girdi/Çıktı 1:

Bir sayı giriniz: 5

6

\*\*\*\*\*\*

3

\*\*\*

2

\*\*

2

\*\*

4

\*\*\*\*

en küçük : 2

en büyük : 6

sayıların toplamı : 17

Örnek Girdi/Çıktı 2:

Bir sayı giriniz: 4

2

\*\*

2

\*\*

3

\*\*\*

1

\*

en küçük : 1

en büyük : 3

sayıların toplamı : 8

===================

Ödevin yüklenmesi:

Sadece proje klasörü içindeki tek bir .JAVA uzantılı dosyayı yükleyin

Başka formatta (doc, txt gibi) gönderilen ödevler değerlendirilmeye alınmayacak ve SIFIR verilecektir.

Ödevlerini herhangi bir program ile sıkıştırıp (rar, zip...) göndermeyin

Ödevinizi, ödev metnin altında bulunan Gönderim Ekle / ADD Submission butonuna tıkladıktan sonra açılan dosya yükleme alanına .JAVA dosyasını sürükle bırak ile yükleyebilirsiniz. nksiyonlarınızın karakter dizisi (string) dönmelerine dikkat ediniz.