

## **BLM2512 Veri Yapıları ve Algoritmalar Gr.1-2-3**

### **Ödev – 2**

**Konu:** Kelime Merdiveni Oyunu

**Problem:** Kelime merdiveni, verilen anlamlı bir kelimedenden başlayıp, her adımda sadece tek bir harfin değiştirilmesiyle yine anlamlı kelimelerin elde edilmesi şeklinde ilerleyerek hedeflenen kelimeye ulaşılmasını sağlayan bir oyundur. Oyunun her aşamasında kelimenin harf sayısı değişmeden kalmalıdır. Aşağıda “stone” kelimesini “money” kelimesine taşıyan bir kelime merdiveni örneği verilmiştir.

stone → **a**tone → **a**lone → **c**lone → clons → coons → conns → cones → coney → **m**oney

Kaynak kelimedenden hedef kelimeye giden birden fazla kelime merdiveni olabilir. Örneğin, yukarıdaki stone→money örneği için bir başka kelime merdiveni;

stone → store → shore → chore → choke → choky → cooky → cooey → coney → money şeklindedir.

Programınız en kısa kelime merdivenini bulmalıdır. Geçerli kelimeler için ödev ekinde verilen sözlük dosyası (dictionary.txt) kullanılacaktır.

Programınız, kullanıcıdan okunan kaynak ve hedef kelimeleri alarak en kısa merdiveni oluşturmalı ve adımlarını ekrana yazdırmalıdır. Problemin farklı çözüm yöntemleri bulunsa da ödevde kelime merdiveni problemini aşağıda anlatılan şekilde çözmeniz beklenmektedir.

1. Kaynak kelime alınır ve sözlükte aynı uzunlukta olup tek bir harfi farklı olan tüm kelimeler bulunur.
2. Her kelime için ayrı bir “stack” oluşturulur ve ilk önce kaynak kelime, sonra da tek harfi farklı kelime kendi stack’ine push edilir.
3. Bu stack’lerin her biri bir queue’ya enqueue edilir. (Queue’da stack’ler saklanacaktır.)
4. Queue’dan bir eleman(stack) dequeue edilir, en üstteki kelimeye bakılır ve hedef kelimeye eşit olup olmadığı kontrol edilir.
  - a. Eşitse, bu stack en kısa kelime merdiveninden oluşan çözümü içermektedir.
  - b. Değilse, bu kelimedenden tek harfi farklı ve aynı uzunluktaki tüm kelimeler sözlükten bulunur. Her yeni kelime için stack kopyalanır ve her kelime kendi stack’ine push edilir. Bu stackler queue’ya enqueue edilir.
5. Hedef kelimeye erişince veya queue boş olana kadar 4. adım tekrarlanır.

**Dikkat!** Stack’lerde kullandığınız kelimelerin kaydını tutmanız gereklidir. Aksi taktirde sonsuz döngü oluşacaktır.

## Örnek:

Kaynak kelime: “smart” Sözlükte tek harfi farklı ve aynı uzunlukta beş kelime bulunmaktadır. Bunların her biri için bir stack oluşturulup queue’ya enqueue edilirler:

```
=====
| scart | | start | | swart | | smalt | | smarm |
| smart | | smart | | smart | | smart | | smart |
+-----+ +-----+ +-----+ +-----+ +-----+
=====
```

| scart | ← queue’daki ilk stack dequeue edilir. Scart kelimesinden tek harfi farklı ve aynı  
| smart | uzunluktaki tüm kelimeler bulunur (Dikkat: “smart” hariç) ve stack kopyalanarak  
+-----+ yeni kelimelerin push edilmeleriyle yeni stack’ler oluşturulur (bu örnekte 7 adet).

Bu adımda oluşan stack’ler:

```
| scant | | scatt | | scare | | scarf | | scarp | | scars | | scary |
| scart | | scart | | scart | | scart | | scart | | scart | | scart |
| smart | | smart | | smart | | smart | | smart | | smart | | smart |
+-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+
```

Eğer hedef kelimeye erişilmediyse bu stack’ler de queue’ya enqueue edilir. Bu aşamada kuyrukta 11 stack bulunmaktadır.

```
=====
| start | | swart | | smalt | | smarm | | scant | | scatt | | scare | | scarf | | scarp | | scars | | scary |
| smart | | smart | | smart | | smart | | smart | | smart | | smart | | smart | | smart | | smart |
+-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+
=====
```

Kuyruktan bir stack dequeue edilir. Start kelimesinden tek harfi farklı ve aynı uzunluktaki tüm kullanılmayan kelimeler bulunur. Stack kopyalanarak yeni kelimelerin push edilmeleriyle yeni stack’ler oluşturulur (bu örnekte 4 adet).

```
| sturt | | stare | | stark | | stars |
| start | | start | | start | | start |
| smart | | smart | | smart | | smart |
+-----+ +-----+ +-----+ +-----+
```

Eğer hedef kelimeye erişilmediyse oluşturulan stack’ler queue’ya enqueue edilir. Bu adımda kuyrukta 14 stack bulunmaktadır.

Sonuçta ya kelime merdiveni içeren stack elde edilecektir, ya da sözlükte uygun kelime kalmayacağı için yeni stack’ler oluşturulamayacak ve queue tamamen boşalacaktır. Bu durumda kaynak ile hedef kelime arasında bir merdiven olmadığı çıktısı üretilmelidir.

### Ekran Çıktıları:

Girişte verilen kaynak ve hedef kelimeleri string olarak kullanıcıdan (ekrandan) alınız. Sözlük için verilen dictionary.txt dosyasını kullanınız. Algoritmanızın her adımında oluşan queue içeriğini ekrana yazdırınız. En sonda ya bulunan kelime merdivenini ya da bulunamadığına dair hata mesajını ekrana yazdırınız.

Aşağıda verilen girdiler için çıktıları oluşturunuz:

**Kaynak:** dears **Hedef:** fears

**Kaynak:** blue **Hedef:** pink

**Kaynak:** bluw **Hedef:** pink

**Kaynak:** devil **Hedef:** angel

**Kaynak:** heart **Hedef:** heart

**Kaynak:** babies **Hedef:** sleepy

**Kaynak:** train **Hedef:** car

### Ödev Teslimi ile ilgili önemli bilgiler:

Aşağıda verilen bütün bilgileri içeren tek bir doküman hazırlayarak **03.04.2024 saat 23:59'a** kadar online.yildiz.edu.tr adresi üzerinde tanımlı ödev **OgrenciNumarasi.zip** dosyasını yükleyiniz.

### Geç Ödev Yükleme Koşulları :

Herhangi bir nedenle gecikme yaşayan öğrencilerin mağdur olmamaları için **04.04.2024** saat **00:01** ve **00:30** arası geç ödev yükleme alanı aktif edilecektir. Bu alandan yükleme yapan öğrencilerin ödevleri 20 puan kesinti ile 80 üzerinden değerlendirilecektir. 30 dakikadan fazla gecikme yaşayan öğrencilerin ödevleri değerlendirilmeyecektir.

### Teslim Edilecekler:

1. Anlatılan problemi programı **C** dilinde yazarak **OgrenciNumarasi.c** dosyasını yükleyiniz.
2. Ekran çıktılarını içeren dosyayı **OgrenciNumarasi.pdf** formatında yükleyiniz.
3. Kısa bir **video** (5-10 dk.) hazırlayınız. Video içeriğinde problemi, geliştirdiğiniz çözümü ve kodunuzu anlatınız. Ardından farklı girdiler ile programınızın çalışmasını gösteriniz. Video linkini raporunuza ekleyiniz. Video linkini public paylaşmayınız, kopyaya sebep olmaktadır.

**TESLİM EDİLECEK BELGELER İLE İLGİLİ DETAY BİLGİYİ CLASSROOM'DA PAYLAŞILAN ODEV\_KURALLARI.RAR DOSYASINDA BULABİLİRSİNİZ:**

### **Teslim Edilecek Dokümanlar:**

- HW#\_OgrenciNumarasi.zip (Örn: HW1\_22011001.zip)
  - OgrenciNumarasi.pdf (Örn: 22011001.pdf)
    - Uygulama video linki
  - OgrenciNumarasi.c (Örn: 22011001.c)

- E-POSTA ile GÖNDERİLEN CEVAPLAR KESİNLİKLE DEĞERLENDİRİLMEYECEKTİR.
- DOSYA DIŞINDA SİSTEME HERHANGİ BİR **DRIVE LINK**'i EKLEMİYİNİZ.
- BAŞKA BİR ÖDEVE VEYA İNTERNET ÜZERİNDE BULUNAN BİR ÇÖZÜME BENZERLİĞİ YÜKSEK OLAN ÖDEVLER KOPYA OLARAK DEĞERLENDİRİLECEKTİR.