



Veri Yapıları ve Algoritmalar 2. Ödev
Kelime Merdiveni

Öğrenci Adı: Mehmet Ali Duran

Öğrenci Numarası: 21011090

Dersin Eğitmeni: Mehmet Amaç Güvensan

Video Linki: <https://youtu.be/Hnz3FIJOyWc>

1. Problem Tanımı:

Bu ödevde verilen kaynak ve hedef kelime arasında sadece bir harf değişerek kelimelerden bir kelime merdiveni oluşturması istenmektedir. Geçerli kelimeler için "dictionary.txt" dosyası verilmiştir. Eğer böyle bir yol var ise ekrana basılmalı yok ise de olmadığına dair bilgi verilmelidir. Kaynak ve hedef kelime kullanıcıdan alınır. Çözüm için stack yapısı ve bu stackleri tutan bir queue yapısı oluşturulması istenmektedir.

2. Problemin Çözümü:

Verilen tanıma göre ilk olarak basit bir şekilde stack ve queue yapısı oluşturulmalıdır. İki yapı için de linked list yapısı kullanılmıştır. Bu yapıları yönetmek için push, pop, peek, enqueue, dequeue fonksiyonları kodlanmıştır. Yardımcı fonksiyonlar ve yapılar eşliğinde ana işlemler main fonksiyonunda koşulmuştur. Kullanıcıdan alınan kaynak kelime en az iki harfli olmalı ve sözlük listesinde bulunmalıdır. Ayrıca hedef kelime ile kaynak kelimenin uzunluğu aynı olmalıdır. Verilen sözlük listesinde bulunan en az iki harfli ve de kaynak ve hedef kelimenin aynı olduğu geçerli kelimeler alınır. Geçerli format girilene kadar sorulmaya devam edilir. Artık kaynak ve hedef kelime elde edilmiştir. Kaynaktan hedefe giden bir harfi farklı kelimeler bulunmalıdır. Bu kelimeleri aramak için her defasında sözlüğü tamamen yüklemek ve bütün sözcüklerde aramak yerine sözlük listesinden sadece kaynak kelime uzunluğunda olan kelimeler programa aktarılmıştır ve arama uzayı küçültülerek yaklaşık 10 kat daha az işlem yapılması sağlanmıştır. Daha sonra kaynak kelimedenden bir harfi farklı olan kelimeler elde edilir. Elde edilen kelime sayısı kadar stack e ilk olarak kaynak kelime daha sonra bir harfi farklı olan kelimeler eklenir bu stackler de tek tek queue ya eklenir. Queue dan bir stack dequeue edilir ve stack den peek fonksiyonu ile en üstteki kelime alınır. Aranan hedef kelimeye eşitse bulunmak istenen yol bulunmuştur ve döngü sonlanabilir. Eğer eşitlik yoksa bu yeni kelimedenden bir harfi farklı diğer kelimeler bulunur, bu kelimelerin ise daha önce kullanılmamış olması gerekir. Bunun için bir tane kullanılan kelimeler listesi tutulmuştur. Yeni kelimeler ile de kelime sayısı kadar(daha önce kullanılmayan kelimeler için) yeni stack oluşturulur ve hedef kelimeye giden yol aranmaya devam edilir. Olası bütün kelimeler bakıldığında ya aranan kelime bulunmuş ve döngü sonlanıp ekrana basılmıştır ya da böyle bir yol olmadığı için queue boşalıp kullanıcıya böyle bir yol yoktur bilgisi verilmiştir.

3. Karşılaşılan Sorunlar:

Karşılaşılan sorunlardan biri kelimenin uzunluğu belli olmasına rağmen bütün kelimeler içinde arama yapılmasıydı bu da çok fazla gereksiz işlem demek. Çözüm olarak sadece kaynak kelime uzunluğunda olan kelimeler bir listeye çekildi ve o liste içinde arama yapıldı.

Diğer bir sorun stack e push yaparken eklemesi kolay olduğu için listenin önüne ekleme yapılacak şekilde tasarlandı. Fakat bu daha sonra stack i kopyalayıp yeni kelime de eklenerek queue ya enqueue edileceği sırada stack ters dönüyor ve programı bozuyordu. Bunun çözümü için ilk olarak ara bir stack e push edip bu ara stack i que ya enqueue edince sorun çözüldü.

Ekran Çıktıları:

dears kelimesi fears kelimesinden bir harf farklı. O yüzden sadece iki kelime var.

```
C:\Code Practices\Data Struct x + v
Enter the source word: dears
Enter the target word: fears
dears-> fears-> Top Of The Stack

Total word number: 2
-----
Process exited after 3.082 seconds with return value 0
Press any key to continue . . .
```

blue kelimesinden pink kelimesine giden yol:

```
Enter the source word: blue
Enter the target word: pink
blue-> blub-> blob-> boob-> book-> bonk-> bink-> pink-> Top Of The Stack

Total word number: 8
-----
Process exited after 6.211 seconds with return value 0
Press any key to continue . . .
```

bluw kelimesi sözlükte olmadığı için arama yapmıyor:

```
C:\Code Practices\Data Struct x + v
Enter the source word: bluw
Please enter valid word(The word you entered is not in the dictionary
Enter the source word: |
```

devil -> angel:

```
Enter the source word: devil
Enter the target word: angel
devil-> devel-> level-> lever-> leger-> luger-> auger-> anger-> angel-> Top Of The Stack

Total word number: 9
-----
Process exited after 55.77 seconds with return value 0
Press any key to continue . . .
```

heart -> heart örneğinde heart kelimesi ilk başta kullanılan kelimeler listesine eklendiği için diğer heart kelimesi aranamıyor. Yani yol yoktur:

```
Enter the source word: heart
Enter the target word: heart

There is not kind of path
-----
Process exited after 7.406 seconds with return value 0
Press any key to continue . . .
```

babies -> sleepy:

```
Enter the source word: babies
Enter the target word: sleepy
babies-> gabies-> gables-> gabled-> sabled-> sailed-> stiled-> stiles-> steles-> stells->
steels-> steeps-> sleeps-> sleepy-> Top Of The Stack

Total word number: 14
-----
Process exited after 8.81 seconds with return value 0
Press any key to continue . . .
```

train kelimesi car ile aynı uzunlukta olmadığı için aramaya izin vermiyor:

```
Enter the source word: train
Enter the target word: car
Enter same length word
Enter the target word:
```