

Veri Yapıları ve Algoritmalar 1. Ödev Raporu

Öğrenci Adı: Ertuğrul Şentürk

Öğrenci No: 18011028

Eğitmen: Doç. Dr. M. Elif KARSLIGİL

Açıklama Bölümü

Fonksiyon Tanımlamaları:

- 1- Linkli liste yapısında previous next nodları adres ve value bilgileri saklandı.
- 2- create_value fonksiyonu verilen değerlere göre yeni bir linkli liste elemanı oluşturmak için oluşturuldu. Boş olarak oluşturulan bir versiyonu da overload edildi.
- 3- push_back fonksiyonu verilen linkli listenin sonuna verilen yeni elemanı eklemek için oluşturuldu.
- 4- push_front fonksiyonu verilen linkli listenin sonuna verilen yeni elemanı eklemek için oluşturuldu.
- 5- pop_back verilen listenin son elemanını silip değer olarak o elemanı döndürmek için oluşturuldu.
- 6- print_list fonksiyonu verilen linkli listeyi yazdırmak için oluşturuldu.
- 7- size_calculator fonksiyonu verilen linkli listenin boyutunu ölçmek için oluşturuldu.
- 8- get_numbers fonksiyonu ile dosyadan okunan ilk satırdan T ve L değerlerini integer olarak kaydetmek için oluşturuldu.
- 9- startup_text açılış yazısını yazdırmak için oluşturuldu.
- 10- clear_list linkli listeyi temizlemek için oluşturuldu.
- 11- clear_question linkli listeyi temizleme uyarısı vermek için kullanıldı.
- 12- prime fonksiyonu ve list modifier fonksiyonu istenilen ana işlemleri yapmak için oluşturuldu.

Algoritma:

- 1- Kullanıcıdan veri girişi yöntemi input olarak alındı.
- 2- Q imputu alındığında program sonlandırıldı. 1 ve 2 inputları için program while döngüsüne sokuldu.
- 3- Klavyeden input alınması halinde kullanıcıdan T L ve adresler sırası ile input olarak alındı.

- 4- Adresler text_long adından uzun bir stringe kaydedildi ve o string aradaki boşluk durumlarına göre kelimelere bölündü.
- 5- Dosyadan input alınması halinde get_numbers fonksiyonu ile T ve L değerleri integer olarak ilk satırdan alındı kalan satırlardaki elemanlar aralarındaki boşluklara göre teker teker okundu.
- 6- Her iki metoda göre de alınan adres değerleri geçici bir stringe kaydedildi.
- 7- Linkli listenin boş olması durumunda yeni oluşturulan linkli liste elemanı ilk liste değerine atandı.
- 8- Linkli boş değilse input olarak alınan adres değeri linkli listenin ilk elemanından başlayarak karşılaştırma yapıldı.
- 9- Eğer adres linkli listede yoksa linkli listenin ilk elemanı olarak push_front fonksiyonu ile yeni eleman eklendi.
- 10- Eğer adres linkli listede varsa var olan linkli liste elemanının değeri arttırıldı.
- 11- Arttırılan value değeri input olarak alınan size değeri ile kontrol edildi eğer limit değerinden büyükse o eleman linkli listenin bulunduğu konumundan silinerek linkli listenin başına taşındı.
- 12- Yapılan işlemler sonucunda linkli listenin boyutu kontrol edildi. Eğer linkli listenin boyutu size değerinden büyükse linkli listenin son elemanı silindi.
- 13- Tüm işlemler tamamlandıktan sonra linkli liste yazdırıldı.
- 14- Linkli liste yazdırıldıktan sonra kullanıcıdan input alınarak linkli listenin silinip silinmeyeceği soruldu. Eğer kullanıcı 'Y' inputunu girerse linkli liste clear list fonksiyonu ile silindi ve metot değeri lopta kalınacak şekilde ayarlandı. Bu sayede tekrardan kullanıcıdan input alınması sağlandı.
- 15- Kullanıcı farklı bir input girdiğinde while döngüsünden çıkılarak programın sonlandırılması için metot değeri değiştirildi.

Ekran Görüntüleri:

Case1:

Klavyeden Input

```
--Veri Yapilari 1. Odevi--
1-) Klavye ile giris yapin.
2-) Dosyadan okuyarak giris yapin.
Q-) Cikmak icin Q ya basin.
Veri giris metodunu seciniz: Klavyeden okuma metodu secildi.
Adresi Listenin Basina Koymak Icin Gereken Siniri Giriniz: 2
Maximum Liste Boyutunu Giriniz: 3
Elemanlari Arada Bosluk Birakarak Giriniz ve Enter a basiniz.
AB BA CY AB CY XYZ BA XYZ BA
AB,1
BA,1 <-> AB,1
CY,1 <-> BA,1 <-> AB,1
CY,1 <-> BA,1 <-> AB,2
CY,2 <-> BA,1 <-> AB,2
XYZ,1 <-> CY,2 <-> BA,1
XYZ,1 <-> CY,2 <-> BA,2
XYZ,2 <-> CY,2 <-> BA,2
BA,3 <-> XYZ,2 <-> CY,2
Cache Listesini Silip Yeniden Baslamak Istermisiniz?(Y/N)
```

Dosyadan Input

```
--Veri Yapilari 1. Odevi--
1-) Klavye ile giris yapin.
2-) Dosyadan okuyarak giris yapin.
Q-) Cikmak icin Q ya basin.
Veri giris metodunu seciniz: Dosyadan okuma metodu secildi.
Dosya adini uzantisiz olarak giriniz.
input
AB,1
BA,1 <-> AB,1
CY,1 <-> BA,1 <-> AB,1
CY,1 <-> BA,1 <-> AB,2
CY,2 <-> BA,1 <-> AB,2
XYZ,1 <-> CY,2 <-> BA,1
XYZ,1 <-> CY,2 <-> BA,2
XYZ,2 <-> CY,2 <-> BA,2
BA,3 <-> XYZ,2 <-> CY,2
Cache Listesini Silip Yeniden Baslamak Istermisiniz?(Y/N)
```

Case2:

Klavyeden Input

```
--Veri Yapilari 1. Odevi--
1-) Klavye ile giris yapin.
2-) Dosyadan okuyarak giris yapin.
Q-) Cikmak icin Q ya basin.
Veri giris metodunu seciniz: Klavyeden okuma metodu secildi.
Adresi Listenin Basina Koymak Icin Gereken Siniri Giriniz: 3
Maximum Liste Boyutunu Giriniz: 4
Elemanlari Arada Bosluk Birakarak Giriniz ve Enter a basiniz.
A B A AA BBB B A AB A B A BB
A,1
B,1 <-> A,1
B,1 <-> A,2
AA,1 <-> B,1 <-> A,2
BBB,1 <-> AA,1 <-> B,1 <-> A,2
BBB,1 <-> AA,1 <-> B,2 <-> A,2
BBB,1 <-> AA,1 <-> B,2 <-> A,3
AB,1 <-> BBB,1 <-> AA,1 <-> B,2
A,1 <-> AB,1 <-> BBB,1 <-> AA,1
B,1 <-> A,1 <-> AB,1 <-> BBB,1
B,1 <-> A,2 <-> AB,1 <-> BBB,1
BB,1 <-> B,1 <-> A,2 <-> AB,1
Cache Listesini Silip Yeniden Baslamak Istermisiniz?(Y/N)
```

Dosyadan Input

```
--Veri Yapilari 1. Odevi--
1-) Klavye ile giris yapin.
2-) Dosyadan okuyarak giris yapin.
Q-) Cikmak icin Q ya basin.
Veri giris metodunu seciniz: Dosyadan okuma metodu secildi.
Dosya adini uzantisiz olarak giriniz.
input
A,1
B,1 <-> A,1
B,1 <-> A,2
AA,1 <-> B,1 <-> A,2
BBB,1 <-> AA,1 <-> B,1 <-> A,2
BBB,1 <-> AA,1 <-> B,2 <-> A,2
BBB,1 <-> AA,1 <-> B,2 <-> A,3
AB,1 <-> BBB,1 <-> AA,1 <-> B,2
A,1 <-> AB,1 <-> BBB,1 <-> AA,1
B,1 <-> A,1 <-> AB,1 <-> BBB,1
B,1 <-> A,2 <-> AB,1 <-> BBB,1
BB,1 <-> B,1 <-> A,2 <-> AB,1
Cache Listesini Silip Yeniden Baslamak Istermisiniz?(Y/N)
```

Akış Diyagramı

