



# VERİ YAPILARI VE ALGORİTMALAR ÖDEV-1

17011701 Onur KOÇKAN

## PROGRAMIN ADIM ADIM AÇIKLANMASI VE YÜRÜTÜLMESİ:

Programda main fonksiyonu dahil total olarak 8 fonksiyon bulunmaktadır bunlar(main hariç);

1- **int sayacSorgula(struct node \*dugum):**düğümde bulunan sayaç(counter) değişkeninin değerini sorgulama

2- **void yazdir():**linkli listedeki bütün elemanları ekrana yazdırma

3- **struct node\* findNode(char adres[20]):**düğümde bulunan adres değişkenini parametre olarak alıp o adrese ait bir node bulunuyormu diye sorgulama

4- **void delete():**linkli listenin en sonunda bulunan node'u silme

5- **int gezinti():**tüm nodeları gezip eleman sayısını döndürme

6- **void deleteAll():**linkli listedeki tüm nodeları silme

7- **void insert(char adres[20],int esik,int buffer):**linkli listeye belirtilen doğrultularda node ekleme

- Programın akışı main fonksiyonunda başlarken Cache Buffer ve Eşik Limitleri Kullanıcıdan Alınır.
- Daha Sonra İşlemlerin Devamında File ilemi Yoksa Console İlemi Devam Edileceği Tercihi Kullanıcıdan Alınır.
- File Tercihinin Seçilmesi Durumunda:input.txt Formatındaki Metin Belgesinin İçerisinde Bulunan Adres Bilgileri Satır Satır Okunur Ve İşleme Alınır.Programın Sonuna Gelindiğinde Kullanıcı Bütün Değerleri Görür Ve İsteği Doğrultusunda Cache Bufferi Tamamıyla Sildirir.

- Console Tercihinin Seçilmesi Durumunda:Kullanıcı İlk Olarak Kaç Adet Adres Giriceğini Belirler Ve Sırasıyla Adresleri Girmeye Başlar.Her Adımda Cache Bufferin Güncel Halini Görme İmkânında Sahiptir.Programın Sonuna Gelindiğinde Kullanıcı Bütün Değerleri Görür Ve İsteği Doğrultusunda Cache Bufferi Tamamıyla Sildirir.

## EKRAN GÖRÜNTÜLERİ:

### Örnek1(Console)

```
C:\Users\Onur Koçan\Desktop\YT\4.DİNEM\Veri Yapılar\ODEV 1\Hw1_17011701\17011701.exe
Buffer Limitini Girin(L):4
Esik Limitini Girin(T):5
File ile İşlem Yapmak İçin(0)
Console ile İşlem Yapmak İçin(1) Giriniz:1
Kac Adres Girilecek?18

Adres Girin:A
  A,1
Adres Girin:B
  B,1
Adres Girin:B      A,1
  B,2
Adres Girin:A      A,1
  B,2
Adres Girin:C      A,2
  C,1      B,2
Adres Girin:C      A,2
  C,2      B,2
Adres Girin:A      A,3
  C,2      B,2
Adres Girin:A      A,4
  C,2      B,2
Adres Girin:A      A,5
  A,6      C,2      B,2
Adres Girin:C      C,3      B,2
Adres Girin:D      A,6      C,3      B,2
  D,1
Adres Girin:E      D,1      A,6      C,3
  E,1
Adres Girin:F      E,1      D,1      A,6
  F,1
Adres Girin:D      E,1      D,2      A,6
  F,1
Adres Girin:G      F,1      E,1      D,2
  G,1
Adres Girin:D      F,1      E,1      D,3
  G,1
Adres Girin:B      G,1      F,1      E,1
  B,1
Tum Liste Silinsinmi? Evet(0) Hayir(1):0
  G,1      F,1      E,1      SILINIYOR...
  F,1      E,1      SILINIYOR...
  E,1      SILINIYOR...
SILINDI!

Cache Buffer Temizlendi!
-----
Process exited after 42.76 seconds with return value 0
Press any key to continue . . .
```

```
C:\Users\Onur Ko\kan\Desktop\YI\4.D\NEM\Veri Yapı\ODEV 1\Hw1_17011701\17011701.exe
Buffer Limitini Girin(L):4
Esik Limitini Girin(T):5
File ile İşlem Yapmak İçin(0)
Console ile İşlem Yapmak İçin(1) Giriniz:1
Kac Adres Girilecek?18

Adres Girin:A
  A,1
Adres Girin:B
  B,1      A,1
Adres Girin:B
  B,2      A,1
Adres Girin:A
  B,2      A,2
Adres Girin:C
  C,1      B,2      A,2
Adres Girin:C
  C,2      B,2      A,2
Adres Girin:A
  C,2      B,2      A,3
Adres Girin:A
  C,2      B,2      A,4
Adres Girin:A
  C,2      B,2      A,5
Adres Girin:A
  C,2      B,2      A,6
Adres Girin:C
  C,2      B,2
Adres Girin:D
  C,3      B,2
Adres Girin:D
  D,1      A,6      C,3      B,2
Adres Girin:E
  E,1      D,1      A,6      C,3
Adres Girin:F
  F,1      E,1      D,1      A,6
Adres Girin:D
  F,1      E,1      D,2      A,6
Adres Girin:G
  G,1      F,1      E,1      D,2
Adres Girin:D
  G,1      F,1      E,1      D,3
Adres Girin:B
  B,1      G,1      F,1      E,1
Tum Liste Silinsinmi? Evet(0) Hayir(1):1

Cache Buffer Temizlenmedi.Son Hali;   B,1      G,1      F,1      E,1
-----
Process exited after 17.11 seconds with return value 0
Press any key to continue . . .
```

## Örnek2(File)

```
Input - Not Defteri
Dosya Düzen Biçim Görünüm Yardım
AB
BA
CY
AB
CY
XYZ
BA
XYZ
BA

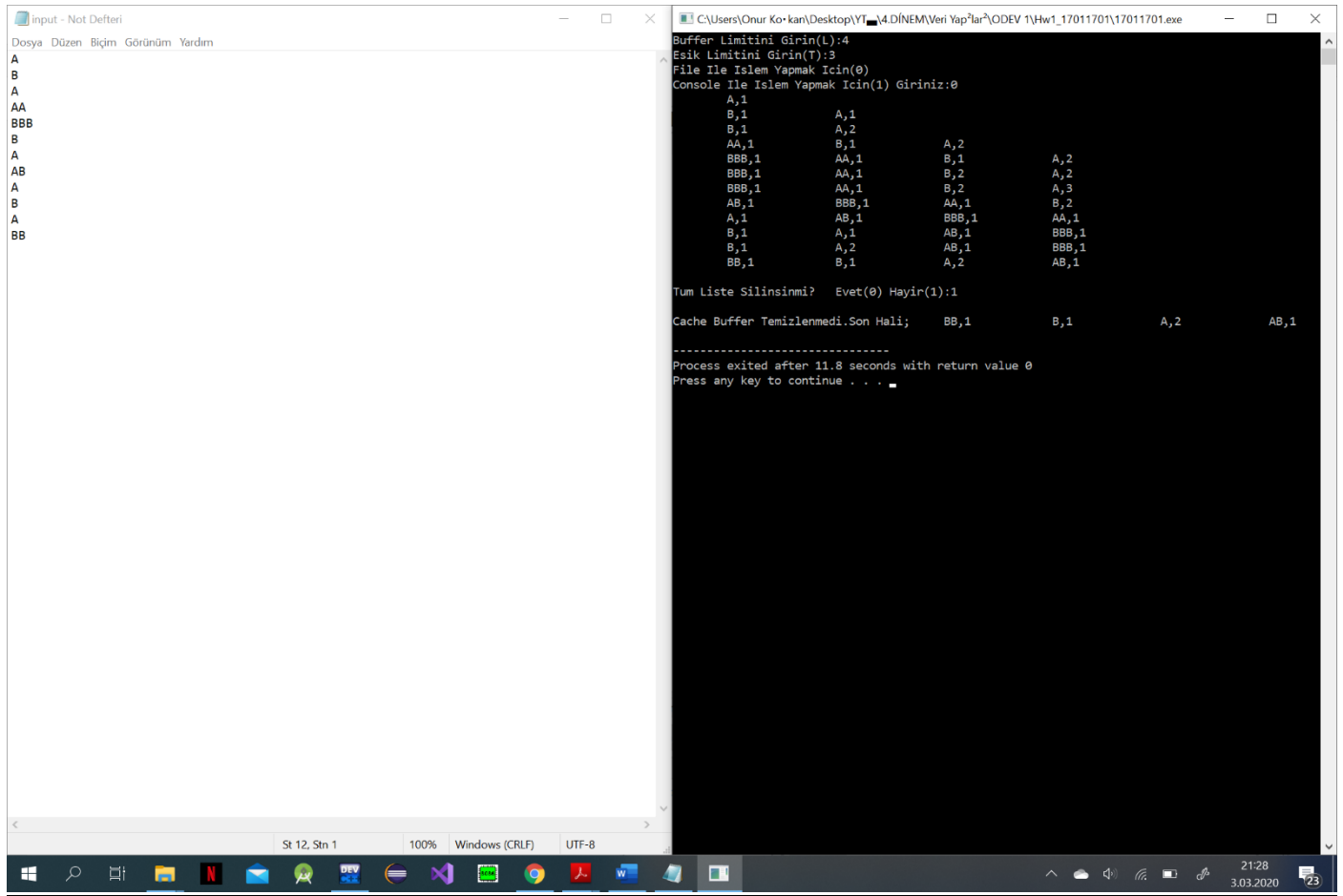
C:\Users\Onur Ko\kan\Desktop\YI\4.D\NEM\Veri Yapı\ODEV 1\Hw1_17011701\17011701.exe
Buffer Limitini Girin(L):3
Esik Limitini Girin(T):2
File ile İşlem Yapmak İçin(0)
Console ile İşlem Yapmak İçin(1) Giriniz:0
  AB,1
  BA,1      AB,1
  CY,1      BA,1      AB,1
  CY,1      BA,1      AB,2
  CY,2      BA,1      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

Tum Liste Silinsinmi? Evet(0) Hayir(1):0
  XYZ,2      CY,2      SILINIYOR...
  CY,2      SILINIYOR...

SILINDI!

Cache Buffer Temizlendi!
-----
Process exited after 48.64 seconds with return value 0
Press any key to continue . . .
```

## Örnek3(File)



```
input - Not Defteri
Dosya Düzen Biçim Görünüm Yardım
A
B
A
AA
BBB
B
A
AB
A
B
A
BB

C:\Users\Onur Ko\kan\Desktop\YI\4.DİNEM\Veri Yapılar\ODEV 1\Hw1_17011701\17011701.exe
Buffer Limitini Girin(L):4
Stack Limitini Girin(T):3
File ile İşlem Yapmak İçin(0)
Console ile İşlem Yapmak İçin(1) Giriniz:0
A,1
B,1
AA,1
BBB,1
AB,1
A,1
B,1
BB,1
A,1
A,2
B,1
AA,1
B,2
AA,1
B,2
AB,1
BBB,1
AA,1
BBB,1
AB,1
A,1
B,1
BB,1
A,1
A,2
AB,1
BBB,1
A,2
AB,1

Tum Liste Silinsinmi? Evet(0) Hayır(1):1
Cache Buffer Temizlenmedi.Son Hali; BB,1 B,1 A,2 AB,1
-----
Process exited after 11.8 seconds with return value 0
Press any key to continue . . .
```

### PROGRAMIN KODLARI:

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
#include <stddef.h>
```

```
struct node{ //double linked list yapımız için gerekli struct tanımlaması
```

```
    int counter;
```

```
    char adres[20];
```

```
    struct node *next;
```

```
    struct node *prev;
```

```
};
```

```
struct node *head=NULL; //structun ilk değeri başta global değişken olarak null atanıyor
```

```
int elemanSayisi=0; //linked listimizin eleman sayisini tutmak için elemanSayisi değişkeni tanımlanıyor.
```

```
int sayacSorgula(struct node *dugum){ //istenilen node ın sayacını sorgular.
```

```
    return dugum->counter;
}
```

void yazdir() //linked list içerisindeki tüm nodeların adres değerlerini sırasıyla yazdırır.

```
{
    struct node* temp = head;
    while (temp != NULL)
    {
        printf("  %s,%d ", temp->adres,temp->counter);
        temp = temp->next;
    }
}
```

struct node\* findNode(char adres[20]){ //istenilen adres parametresi gönderildiği takdirde o adres değişkenine sahip bir node varmı yokmu kontrolünü yapar.

```
    struct node* Node;
    Node=head;
    while(Node!=NULL){
        if(strcmp(Node->adres,adres)==0){ //mevcut node'un adresi=parametre olarak gelen adres
            return Node; //BULUNDU
        }
        Node=Node->next;
    }
    //BULUNAMADI
    return NULL;
}
```

void delete(){ //linkli listemizin en son elemanını siler.

```
    struct node *temp=head;
    while(temp->next!=NULL){
        temp=temp->next;
    }
    temp->prev->next=NULL;
```

```

        free(temp);
    }

int gezinti() { //tüm nodeları gezerek(baştan en son node a kadar) linkli listenin eleman sayısını döndürür.
    struct node *temp=head;
    int say=0;
    while(temp->next != NULL){
        temp = temp->next;
        say++;
    }
    return say;
}

void deleteAll() { // programın son kısmında kullanıcı isteği doğrultusunda tüm linkli listeyi temizleyecek
olan kod parçasığı
    int say=gezinti();
    struct node *gec;
    int i;
    for (i=0;i<say;i++)
    {
        gec = head->next;
        gec->prev = NULL;
        free(head);
        head = gec;
        yazdir();
        printf("SILINIYOR...\n",gec->adres);
    }
    head=head->next;
    free(head);
    yazdir();
    printf("SILINDI!\n",gec->adres);
    return;
}

```

```

void insert(char adres[20],int esik,int buffer){
    if (head==NULL){ //İLK NODE EKLENIRKEN
        struct node* newNode = (struct node*)malloc(sizeof(struct node));
        elemanSayisi++;
        strcpy(newNode->adres,adres);
        newNode->next = NULL;
        newNode->prev = NULL;
        newNode->counter=1;
        head=newNode;
        yazdir();
    }

    else{
        if(findNode(adres)==NULL){ // ELEMEN HIC YOKSA
            if(elemanSayisi<buffer){ //ELEMEN SAYISI BUFFER LIMITINDEN DUSUK ISE
BASA EKLE
                struct node* newNode = (struct node*)malloc(sizeof(struct node));
                elemanSayisi++;
                strcpy(newNode->adres,adres);
                newNode->next = head;
                newNode->prev = NULL;
                newNode->counter=1;
                head->prev=newNode;
                head=newNode;
                yazdir();
            }

            else{//buffer limiti asildi son eleman silinecek,yeni eleman basa eklenicek
                delete();

                struct node* newNode = (struct node*)malloc(sizeof(struct node));
                strcpy(newNode->adres,adres);
                newNode->next = head;
                newNode->prev = NULL;
                newNode->counter=1;
            }
        }
    }
}

```

```
head->prev=newNode;
head=newNode;
yazdir();

}
```

```
}//ELEMAN VARSA
```

```
else{
    struct node* node=findNode(adres);
    if(node->counter==esik){ //ilk esigi gecme
        if(node->prev==NULL){ //node bastaysa
            node->counter++;
            yazdir();
        }
        else if(node->next==NULL){ //node sondaysa
            node->counter++;
            node->prev->next=NULL;
            node->next=head;
            node->prev=NULL;
            head->prev=node;
            head=node;
            yazdir();
        }
        else{ //node ortalaradaysa
            node->counter++;
            node->next->prev=node->prev;
            node->prev->next=node->next;
            node->next=head;
            head->prev=node;
            node->prev=NULL;
            head=node;
            yazdir();
        }
    }
}
```



```

        }

    }

    else{ //esik gecilme problemi yok sayac articaK
        node->counter++;
        yazdir();
    }

}

}

}

```

```

int main(int argc, char *argv[]) {
    int buffer,esik,adet,karar,karar1;
    char adres[50];

    printf("Buffer Limitini Girin(L):");
    scanf("%d",&buffer);
    printf("Esik Limitini Girin(T):");
    scanf("%d",&esik);

    printf("File Ile Islem Yapmak Icin(0)\nConsole Ile Islem Yapmak Icin(1) Giriniz:");
    scanf("%d",&karar1);
    FILE *dosya;
    if(karar1==0){//FILE ISLEMI ILE PROGRAMIN YURUTULMESI
        dosya=fopen("input.txt","r");
        while(!feof(dosya)){

```

```

        fscanf(dosya,"%s",adres);

        insert(adres,esik,buffer);

        printf("\n");

        //printf("\n Adres : %s",adres);

    }

    fclose(dosya);

```

printf("\nTum Liste Silinsinmi?      Evet(0)      Hayir(1):");//TUM LISTENIN  
TEMİZLENMESİ TERCİHİ

```

scanf("%d",&karar);

if(karar==0){

    deleteAll();

    printf("\nCache Buffer Temizlendi!");

    yazdir();

}

else{

    printf("\nCache Buffer Temizlenmedi.Son Hali");

    yazdir();

}

}

```

if(karar1==1){//CONSOLE ILE PROGRAMIN YÜRÜTÜLMESİ

```

    printf("Kac Adres Girilecek?");

    scanf("%d",&adet);

```

```

while(adet!=0){

    printf("\nAdres Girin:");

    scanf("%s",&adres);

    insert(adres,esik,buffer);

    adet--;

}

```

printf("\nTum Liste Silinsinmi?      Evet(0)      Hayir(1):");//TUM LISTENİN  
TEMİZLENMESİ TERCİHİ

```

scanf("%d",&karar);

```

```
    if(karar==0){
        deleteAll();
        printf("\nCache Buffer Temizlendi!");
        yazdir();
    }
    else{
        printf("\nCache Buffer Temizlenmedi.Son Hali;");
        yazdir();
    }

}

return 0;

}
```