



2021-2022 Güz Yarıyılı

Algoritma Analizi

Proje Ödev : Hashing

Rayene Bech

18011115

I. Yöntem

Kullanılan veri yapılar:

- a. Array of Struct: Hash table için:

```
typedef struct hash_table{  
    int id;  
    char TC[MAX_LENGTH];  
    int order;  
    int reserved;  
} hash_table;
```

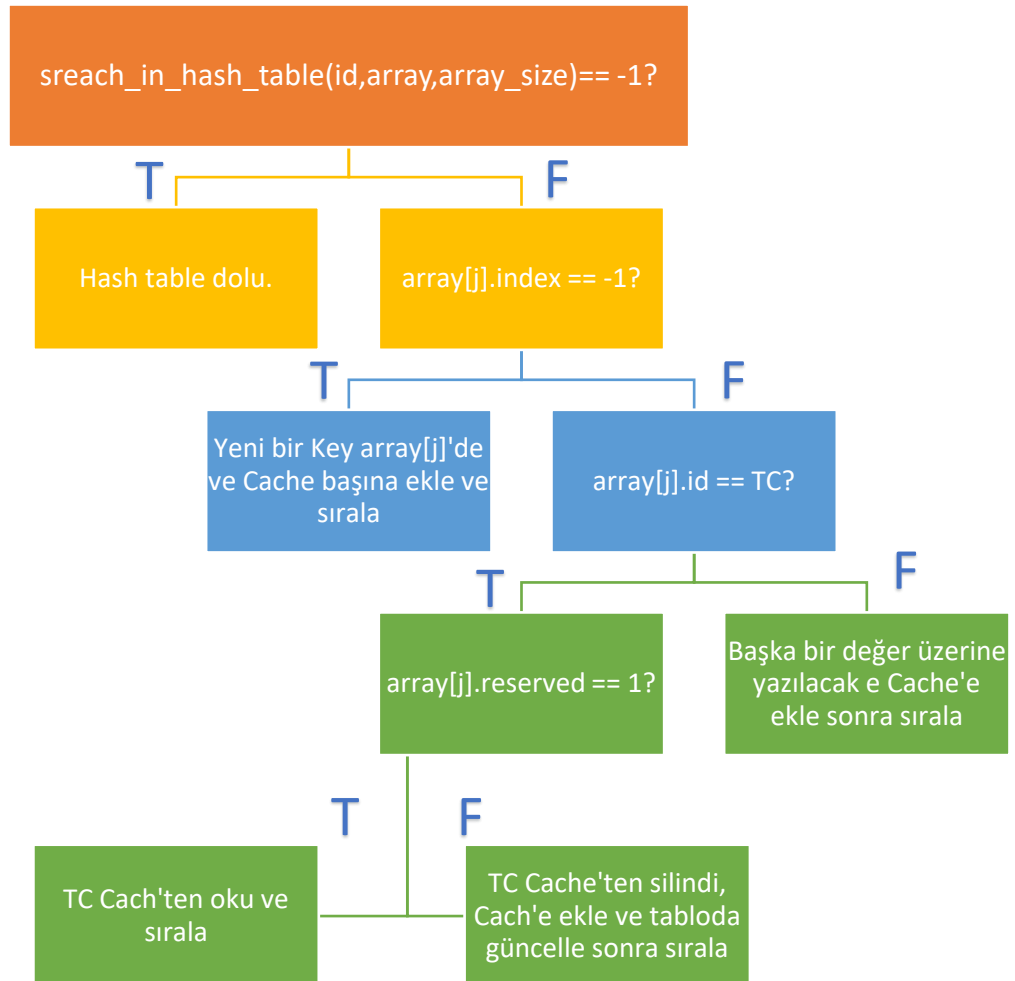
- b. Linked List: Cache bilgileri tutmak için kullanılır

```
struct node {  
    char id[MAX_LENGTH];  
    char name[MAX_LENGTH];  
    char surname[MAX_LENGTH];  
    int birth_year;  
    char address[MAX_LENGTH];  
    struct node* next;  
};  
typedef struct node node;
```

Çözüm Adımları:

1. Program ilk çalıştırıldığında, kullanıcıdan Cache uzunluğu alınır. Onu kullanarak ve Load Factor 0.6 alarak, Hash table uzunluğu **findSize()** fonksiyon yardımıyla hesaplanır. Linked List ve Hash table oluşturulur.
2. Hash table 'deki reserved kısmı bir Flag 'dır. Bir değer Cache 'e ekleyince 1 olur, silince ise 0 olmaktadır. İlk başta 0 ile ilklendirilir.
3. Program kapanmadığı sürece **user_interface()** fonksiyonu sürekli çağırılmaktadır. Bu fonksiyon **TC** numarası kullanıcıdan aldıktan sonra **horner()** fonksiyonu yardımıyla **TC**'ye karşı bir **id** üretir. Bu **id** **sreach_in_hash_table()** fonksiyonla Hash table'de aranmaktadır
4. **Hash table'deki arama işlemi:**
 - c. **Double Hashing** kullanıldığı için, h1 ve h2 iki Hash fonksiyon kullanılır. **Key** değeri 'id' Double Hashing yapınca ona karşı bir indis j elde edilir.
 - d. Double Hashing yaparken bir indis i kullanılır :
$$h(\text{key}, i) = (h1(\text{key}) + i * h2(\text{key})) \% m$$
 - e. Bu işlem durdurmak için 4 farklı durum var:
 - a. Bu indis i tablo uzunluğundan fazla olduysa: aranan **Key yok** + Hash tablosu **dolu**: fonksiyon -1 değer döndürür.

- b. Aranan Key değeri tabloda bulunduysa: ona karşı gelen j indisi döndürülür.
 - c. Boş bir hücre bulunduysa: aranan Key tabloda bulunmuyor ve bu hücreye eklenebilir. $\text{Array}[j].\text{index} = -1$ ilk olan j indisi değeri döndürülür
 - d. Tarama yaparken dolu hücrenin reserved Flag 'ı 0 bulunduysa (Ve değeri Key 'e eşit değilse): Bu değer silindi demektir. Fakat aradığımız Key değeri ondan sonra da bulunabilir. O yüzden arama işlemi burada doldurulmaz. Bu silinen hücrenin indisi j 'de tutulur. Yeni bir indisi k kullanarak tablo uzunluğuna kadar Key değeri aramaya devam edilir. Bulunursa k döndürülür, aksi takdirde j döndürülür ve silinen değer üstüne yazılır.
5. Hash tablosunda arama işlemi bittikten sonra döndürülen değere göre bu durumlar bulunmaktadır:



6. Cache'ten okuma veya yeni deęer eklendięinde, bu deęer Cache'in bařına eklenir:

- Eęer bu deęer ilk defa Cach'e eklenecekse, bütn Cache'tekiler Hash tablosundaki "order" deęer +1 olacak
- Eęer bu deęer Cache'te zaten varsa ve sadece okuma yapılacaksa, sadece Cache'teki Bu deęerden nce bulunan deęerlerin "order" +1 olacak. Ondan sonra bulunan deęerler "order" deęeri deęiřmeyecek.

Zaman Karmařıklığı:

Cache'te Arma:

Hash tablosundaki "order" deęere kadar Cache'te gezilir. En kt durumda Aranan eleman Cache'in en son eleman olması. O halde $O(n)$ olacaktır. En iyi durumda, Cache'in en son gezilen elemanın yine aranırsa, $O(1)$ olacaktır. (Bu da zaten Cache'i kullanmak amacımız)

Cache'e eleman Ekleme:

Eleman eklerken Cache'in bařına eklenir. O yzden $O(1)$ olmaktadır.

Cache'ten elemen silme:

Cache'ten sadece dolu olduęu durumda en son eleman silinir. En son elemana ulařmak iin yine $O(n)$ srmektedir.

Yer Karmařıklığı:

N maksimum uzunluęa sahip olan bir Linked List kullanıldıęı iin yer karmařıklığı $O(N)$ olacaktır.

II. Uygulama

- Kullanıcıdan cache uzunluęu input olarak aldıktan sonra, hash table uzunluęu da hesaplanır. Bu rnekte Cache size = 4 iin Hash_table size =7

```
Please enter the cache size: 4
Array_size is : 7
Printing Hashtable:
TC | order | reserved
x | 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
```

- Ondan sonra TC numarası kullanıcıdan istenmektedir. Horner kurallarına göre bir ID hesaplanır ve double Hashing kullanarak tabloya eklenir (eğer zaten tabloda bulunmamaktadır). Aynı zamanda Cache başına da eklenir :

```
Please enter the TC number of the person: 33445

Information of the newPerson retrieved from the file:
33445, AÇELYA, ŞENLİK, 1990, adana
This Id does not exist in the cache neither in the hash table

Printing the cache:

id: 33445 name AÇELYA, surname: ŞENLİK birthyear 1990 address: adana
|

Printing Hashtable:
TC | order | reserved
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
33445 | 0 | 1
```

- Birkaç kişi cache'e ekledikçe hash tablosundaki "order" değeri değişir :

```
Please enter the TC number of the person: 12345

Information of the newPerson retrieved from the file:
12345, ŞABAN, DEMİRHAN, 1993, istanbul
This Id does not exist in the cache neither in the hash table

Printing the cache:

id: 12345 name ŞABAN, surname: DEMİRHAN birthyear 1993 address: istanbul
|

id: 33445 name AÇELYA, surname: ŞENLİK birthyear 1990 address: adana
|

Printing Hashtable:
TC | order | reserved
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
12345 | 0 | 1
| 0 | 0
33445 | 1 | 1
```

```
Please enter the TC number of the person: 43321

Information of the newPerson retrieved from the file:
43321, ZÜBEYDE, HARMANBAŞI, 2001, izmir
This Id does not exist in the cache neither in the hash table

Printing the cache:

id: 43321 name ZÜBEYDE, surname: HARMANBAŞI birthyear 2001 address: izmir
|

id: 12345 name ŞABAN, surname: DEMİRHAN birthyear 1993 address: istanbul
|

id: 33445 name AÇELYA, surname: ŞENLİK birthyear 1990 address: adana
|

Printing Hashtable:
TC | order | reserved
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
| 0 | 0
12345 | 1 | 1
43321 | 0 | 1
33445 | 2 | 1
```

- Cache'de bulunan bir değeri araniyorsa o değer en başa alınır:

```

Please enter the TC number of the person: 12345
The value already exists, reading from cache:
id: 12345 name ŞABAN, surname: DEMİRHAN birthyear 1993 address: istanbul
Printing Hashtable:
TC | order | reserved
♦ | 0 | 0
  | 0 | 0
  | 0 | 0
  | 0 | 0
12345 | 0 | 1
43321 | 1 | 1
33445 | 2 | 1

Printing the cache:

id: 12345 name ŞABAN, surname: DEMİRHAN birthyear 1993 address: istanbul
|

id: 43321 name ZÜBEYDE, surname: HARMANBAŞI birthyear 2001 address: izmir
|

id: 33445 name AÇELYA, surname: ŞENLİK birthyear 1990 address: adana
|

```

- Cache dolu olduğunda yeni bir eleman eklendiğinde, Cache'teki en son değeri silinir ve Hash tablosu da güncellenir (reserved flag' 0 yaparak)

```

Please enter the TC number of the person: 43213

Information of the newPerson retrieved from the file:
43213, AYSU, ŞATIROĞLU, 1991, istanbul
This Id does not exist in the cache neither in the hash table

The cache is full so delete the last one
The Id to be delted is: 33445
Printing the cache:

id: 43213 name AYSU, surname: ŞATIROĞLU birthyear 1991 address: istanbul
|

id: 32145 name TEYFİK, surname: ALBEN birthyear 1984 address: ankara
|

id: 12345 name ŞABAN, surname: DEMİRHAN birthyear 1993 address: istanbul
|

id: 43321 name ZÜBEYDE, surname: HARMANBAŞI birthyear 2001 address: izmir
|

Printing Hashtable:
TC | order | reserved
j | 0 | 0
32145 | 1 | 1
43213 | 0 | 1
  | 0 | 0
12345 | 2 | 1
43321 | 3 | 1
33445 | 3 | 0

```