



Homework: **Cache Buffer**

Name-Surname: **Burak Atalay**

Number: **22011641**

Eğitmen: **Göksel Biricik**

Problem Tanımı:

Kullanıcıdan alınan adresler ile bir cache buffer gerçekleştirilmesi isteniyor. Bunun doubly linked list veri yapısı kullanarak gerçekleştirilmesi isteniyor. Cache buffer'ın yapması gereken görevler şunlar: Yeni adres girilince eğer o adrese sahip node yoksa yeni bir node yaratılıp head yapılır. Eğer ki varsa o node'un sayacı artırılır ve eşik değerden yüksek olursa bu node head yapılır. Son olarak yeni node eklenirken buffer kapasitesini aşacaksa kuyruktaki node silinir.

Problemin Çözümü:

Önce buffer'ın kapasitesi ve her bir node için geçerli olan eşik değeri kullanıcıdan alınır. Sonra kullanıcıya dosyadan mı yoksa konsoldan mı girdi olacağı sorulur. Alınan her bir girdi için readInput fonksiyonu çağrılır. Bu fonksiyonda girdi için buffer'da linear arama yapan search fonksiyonu çağrılır. Search fonksiyonundan null dönerse bu girdiye sahip node olmadığından yeni bir node yaratılmaya girişilir. Ancak öncesinde buffer'ın size'ı kapasitesine eşit mi diye bakılır ve eğer öyleyse buffer'ın tail pointer'ı tail'in gösterdiği node silinerek güncellenir. Eğer ki kullanıcı program başında kapasiteyi 1 vermişse tail güncelleme kısmında buffer boş hale geleceğinden buffer'ın head'i de null yapılır. Tail güncelleme sonrası buffer'ın size'ı 1 azaltılır. Sonrasında kapasite aşılma durumu için girdiğimiz bu if bloğundan çıkılır ve bu bloğa girip girmemizden bağımsız olarak yeni bir node yaratılır ve buffer'ın head'i bu node yapılır. Eğer ki bu node yaratılmadan önce buffer boş haldeyse yani head null ise tail'e de bu yaratılan node atanır. Bu işlem sonunda buffer'ın size'ı 1 artırılır.

Eğer ki search fonksiyonundan null dönmemişse dönen pointer kullanıcının girdiği adrese sahip olan node'dur. Bu sebeple bu node'un counter'ını 1 artırıp eşik değeri geçti mi diye kontrol edilir. Eğer geçmişse ve buffer'ın head'i değilse bu node buffer'ın yeni head'i yapılır. Bunu yaparken şöyle bir edge case olabilir: Eğer bu node buffer'ın tail'i ise o zaman bu işlem sırasında tail'in güncellenmesi gerekir.

Kullanıcı herhangi bir anda buffer'ı silme seçeneğine sahiptir. Bu durumda buffer free'lenir ve aynı kapasite ve eşik değere sahip yeni bir buffer oluşturulur.

Karşılaşılan Sorunlar

İlk başta doubly linked list implementasyonunda tail pointer'ı tutmamıştım ve bu sebeple $O(n)$ 'lik search işleminden sonra buffer'ın son elemanı ile işlem yapacaksam tekrardan sonuna kadar ilerlemem gerekiyordu yani readInput'un içinde $O(n)$ 'lik bir iş daha gerekiyordu. Asimptotik analiz için bir şey farketmiyor tabii ancak tail pointer'ı tutmak küçük inputlar için gözle görülür bir hız kazandırıyor.

Ekran Görüntüleri

Senaryo 1 (dosyadan)

kapasite = 4, eşik değeri = 5

A B B A C C A A A A C D E F D G D B

```
threshold value for each node: 5
buffer capacity: 4

enter 1 to give addresses from file...
enter 0 to give them on console...
choice: 1
filename: input.txt
A - 1
B - 1 <--> A - 1
B - 2 <--> A - 1
B - 2 <--> A - 2
C - 1 <--> B - 2 <--> A - 2
C - 2 <--> B - 2 <--> A - 2
C - 2 <--> B - 2 <--> A - 3
C - 2 <--> B - 2 <--> A - 4
C - 2 <--> B - 2 <--> A - 5
A - 6 <--> C - 2 <--> B - 2
A - 6 <--> C - 3 <--> B - 2
D - 1 <--> A - 6 <--> C - 3 <--> B - 2
E - 1 <--> D - 1 <--> A - 6 <--> C - 3
F - 1 <--> E - 1 <--> D - 1 <--> A - 6
F - 1 <--> E - 1 <--> D - 2 <--> A - 6
G - 1 <--> F - 1 <--> E - 1 <--> D - 2
G - 1 <--> F - 1 <--> E - 1 <--> D - 3
B - 1 <--> G - 1 <--> F - 1 <--> E - 1

-----
Process exited after 15.12 seconds with return value 0
Press any key to continue . . . |
```

Senaryo 2 (dosyadan)

kapasite = 3, eşik değeri = 1

google.com youtube.com sahibinden.com google.com sahibinden.com getir.com
youtube.com getir.com youtube.com

```
threshold value for each node: 1
buffer capacity: 3

enter 1 to give addresses from file...
enter 0 to give them on console...
choice: 1
filename: input5.txt
google.com - 1
youtube.com - 1 <-> google.com - 1
sahibinden.com - 1 <-> youtube.com - 1 <-> google.com - 1
google.com - 2 <-> sahibinden.com - 1 <-> youtube.com - 1
sahibinden.com - 2 <-> google.com - 2 <-> youtube.com - 1
getir.com - 1 <-> sahibinden.com - 2 <-> google.com - 2
youtube.com - 1 <-> getir.com - 1 <-> sahibinden.com - 2
getir.com - 2 <-> youtube.com - 1 <-> sahibinden.com - 2
youtube.com - 2 <-> getir.com - 2 <-> sahibinden.com - 2

-----
Process exited after 11.5 seconds with return value 0
Press any key to continue . . . |
```

Senaryo 3 (konsoldan)

kapasite = 3, eşik değeri = 1

twitter.com google.com yandex.com yandex.com yandex.com yandex.com twitter.com
twitter.com twitter.com google.com google.com google.com github.com github.com
google.com

```
threshold value for each node: 3
buffer capacity: 3

enter 1 to give addresses from file...
enter 0 to give them on console...
choice: 0

address: twitter.com
twitter.com - 1

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

address: google.com
google.com - 1 <-> twitter.com - 1

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

address: yandex.com
yandex.com - 1 <-> google.com - 1 <-> twitter.com - 1

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1
```

```
address: yandex.com
yandex.com - 2 <-> google.com - 1 <-> twitter.com - 1
```

```
enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1
```

```
address: yandex.com
yandex.com - 3 <-> google.com - 1 <-> twitter.com - 1
```

```
enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1
```

```
address: yandex.com
yandex.com - 4 <-> google.com - 1 <-> twitter.com - 1
```

```
enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1
```

```
address: twitter.com
yandex.com - 4 <-> google.com - 1 <-> twitter.com - 2
```

```
enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1
```

```
address: twitter.com
yandex.com - 4 <-> google.com - 1 <-> twitter.com - 3
```

```
enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1
```

```
address: twitter.com
twitter.com - 4 <-> yandex.com - 4 <-> google.com - 1
```

```
enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 2
```

```
address: google.com
google.com - 1
```

```
enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1
```

```

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

address: google.com
google.com - 2

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

address: google.com
google.com - 3

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

address: github.com
github.com - 1 <-> google.com - 3

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

```

```

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

address: github.com
github.com - 2 <-> google.com - 3

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 1

address: google.com
google.com - 4 <-> github.com - 2

enter 1 to continue...
enter 2 to delete buffer...
enter 0 to finish...
choice: 0

```

```

-----
Process exited after 201.4 seconds with return value 0
Press any key to continue . . . |

```

```

- compilation time: 0,248

```