

BLM2512 Veri Yapıları ve Algoritmalar Gr.1-2-3

Ödev – 1

Konu: Cache Buffer (Önbellek Tamponu) Tasarımı

Problem: Bu ödevde, double linked list (ikili linkli liste) veri yapısını kullanarak bir cache buffer sistemi tasarlamanız ve gerçekleştirmeniz istenmektedir. Gerçekleştireceğiniz cache buffer, bir network ortamında yapılan web sayfası isteklerini, hızlı ve kaliteli hizmet verebilmek için depolamak ve servis için önceliklendirmek amacı ile kullanılacaktır.

Ödev 3 ana bölümden oluşmaktadır:

1. **Double linked list düğümlerinin ve veri yapısının gerçekleştirilmesi:**
Her bir düğümde; istenen sayfanın adresi, toplam istek sayısı (sayacı), önceki ve sonraki düğümlerin işaretçileri tutulmalıdır. Düğümler üzerinde gezinti, sayaç sorgulama, düğüm ekleme ve silme gibi işlemler gerçekleştirilmelidir.
2. **Verilen işlemin double linked list kullanılarak çözümü:**
 - a. **Giriş bilgisinin okunması:**
 - i. İstek yapılan sayfaları string olarak okuyunuz (Ör: ABC, ABE, BCD, ...). Sayfalar bir metin belgesinden satır satır da okunabilmelidir.
 - b. **Girdinin işlenmesi:**
 - i. Girilen sayfa (veya dosyadan okunan 1 satır) önceden cache buffer'da yoksa yeni bir düğüm olarak yaratılır, sayacı 1 yapılır ve başlangıç düğümü (head node) olarak cache buffer'a eklenir.
 - ii. Girilen sayfa cache buffer'da mevcut ise, ilgili düğümün sayacı 1 arttırılır.
 1. Sayaç belirlenen bir **eşik değeri**nin (Ör: $T = 10$) üzerine geçerse, bu düğüm cache buffer'da **head node** olarak atanır. (Önceki ve sonraki düğümlerin işaretçileri de düzenlenir.)
 - c. **Cache buffer'ın bakımı:**
 - i. Cache buffer'ın **belirli bir kapasitesi** vardır (Ör: $L=20$). Bu kapasite aşılsa, cache buffer'ın **son elemanı (düğümü)** silinir (2.b maddesinde kontrol edilmelidir).
3. **Cache buffer'ın hizmet vermesi:** Her yeni istek geldiğinde, 2.b maddesi işletilir ve cache buffer'ın mevcut hali linkli liste şeklinde, düğümlerin içerikleri ile (adres ve sayacı) ekranda gösterilir.
4. **Cache Buffer'ın Temizlenmesi:** Kullanıcıya listeleme işleminden sonra cache'i temizlemek isteyip istemediği sorulmalıdır. Temizlemeye onay verilirse listedeki tüm elemanlar silinmelidir.

Örnek:

(Yer tutmaması için girdiler satırlarda değil, yanyana ve birer karakter olarak gösterilmiştir.)

T=5, L=4

İSTEKLER: A B B A C C A A A A C D E F D G D B

Ekran çıktıları: (parantez içindeki kırmızı ifadeler açıklama için eklenmiştir)

A,1 (A geldi, yeni düğüm head node atandı)

B,1 ⇌ A,1 (B geldi, yeni düğüm head node atandı)

B,2 ⇌ A,1 (B geldi, sayacı arttı)

B,2 ⇌ A,2 (A geldi, sayacı arttı)

C,1 ⇌ B,2 ⇌ A,2 (C geldi, yeni düğüm head node atandı)

C,2 ⇌ B,2 ⇌ A,2 (C geldi, sayacı arttı)

C,2 ⇌ B,2 ⇌ A,3 (A geldi, sayacı arttı)

C,2 ⇌ B,2 ⇌ A,4 (A geldi, sayacı arttı)

C,2 ⇌ B,2 ⇌ A,5 (A geldi, sayacı arttı)

A,6 ⇌ C,2 ⇌ B,2 (A geldi, sayacı arttı, [T=5] A,6 head node olarak atandı)

A,6 ⇌ C,3 ⇌ B,2 (C geldi, sayacı arttı)

D,1 ⇌ A,6 ⇌ C,2 ⇌ B,2 (D geldi, yeni düğüm head node atandı)

E,1 ⇌ D,1 ⇌ A,6 ⇌ C,2 (E geldi, yeni düğüm head node atandı [L=4] son (5.) düğüm silindi)

F,1 ⇌ E,1 ⇌ D,1 ⇌ A,6 (F geldi, yeni düğüm head node atandı [L=4] son (5.) düğüm silindi)

F,1 ⇌ E,1 ⇌ D,2 ⇌ A,6 (D geldi, sayacı arttı)

G,1 ⇌ F,1 ⇌ E,1 ⇌ D,1 (G geldi, yeni düğüm head node atandı [L=4] son (5.) düğüm silindi)

G,1 ⇌ F,1 ⇌ E,1 ⇌ D,2 (D geldi, sayacı arttı)

B,1 ⇌ G,1 ⇌ F,1 ⇌ E,1 (B geldi, yeni düğüm head node atandı [L=4] son (5.) düğüm silindi)

Ekran Çıktıları:

Girişte verilen ifadeyi string olarak adım adım veya metin dosyasından bir seferde alınız. Yapılan her istek sonunda cache buffer'ın mevcut linkli liste yapısını, düğümlerin içerikleri ile ekrana yazdırınız.

Ödev Teslimi ile ilgili önemli bilgiler:

Aşağıda verilen bütün bilgileri içeren tek bir doküman hazırlayarak **17.03.2024 saat 23:59'a** kadar online.yildiz.edu.tr adresi üzerinde tanımlı ödev **OgrenciNumarasi.zip** dosyasını yükleyiniz.

Geç Ödev Yükleme Koşulları :

Herhangi bir nedenle gecikme yaşayan öğrencilerin mağdur olmamaları için **18.03.2024** saat **00:01** ve **00:30** arası geç ödev yükleme alanı aktif edilecektir. Bu alandan yükleme yapan öğrencilerin ödevleri 20 puan kesinti ile 80 üzerinden değerlendirilecektir. 30 dakikadan fazla gecikme yaşayan öğrencilerin ödevleri değerlendirilmeyecektir.

Teslim Edilecekler:

1. Anlatılan problemi çözen linkli liste veri yapısını ve ilgili fonksiyonları içeren programı **C** dilinde yazarak **OgrenciNumarasi.c** dosyasını yükleyiniz.
2. Ekran çıktıları içeren dosyayı **OgrenciNumarasi.pdf** formatında yükleyiniz.
3. Kısa bir **video** (5-10 dk.) hazırlayınız. Video içeriğinde problemi, geliştirdiğiniz çözümü ve kodunuzu anlatınız. Ardından farklı girdiler ile programınızın çalışmasını gösteriniz. Video linkini raporunuza ekleyiniz. Video linkini public paylaşmayınız, kopyaya sebep olmaktadır.

TESLİM EDİLECEK BELGELER İLE İLGİLİ DETAY BİLGİYİ CLASSROOM'DA PAYLAŞILAN ODEV_KURALLARI.RAR DOSYASINDA BULABİLİRSİNİZ:

Teslim Edilecek Dokümanlar:

- HW#_OgrenciNumarasi.zip (Örn: HW1_22011001.zip)
 - OgrenciNumarasi.pdf (Örn: 22011001.pdf)
 - Uygulama video linki
 - OgrenciNumarasi.c (Örn: 22011001.c)

- E-POSTA ile GÖNDERİLEN CEVAPLAR KESİNLİKLE DEĞERLENDİRİLMEMEYECİKTİR.
- DOSYA DIŞINDA SİSTEME HERHANGİ BİR **DRIVE LİNK'i** EKLEMİYİNİZ.
- BAŞKA BİR ÖDEV VEYA İNTERNET ÜZERİNDE BULUNAN BİR ÇÖZÜME BENZERLİĞİ YÜKSEK OLAN ÖDEVLER KOPYA OLARAK DEĞERLENDİRİLECEKTİR.