

İşletim Sistemlerine Giriş

Bellek Yönetimi (Memory Management)

Yazılım ile LRU Benzetimi

Donanım kullanmadan LRU algoritmasının yazılım ile gerçekleştirimidir. Kullanılan algoritma NRU(Not Frequently Used) algoritmasıdır.

Her sayfa ile ilgili bir yazılım sayacına ihtiyaç vardır. Sayaçlar başlangıçta sıfırlanır. Her saat kesmesinde işletim sistemi bellekteki tüm sayfaları tarar ve algoritmayı gerçekleştirir.

Her sayfa için $R(0 \mid 1)$ biti, o sayfanın yazılımsal sayacına eklenir.

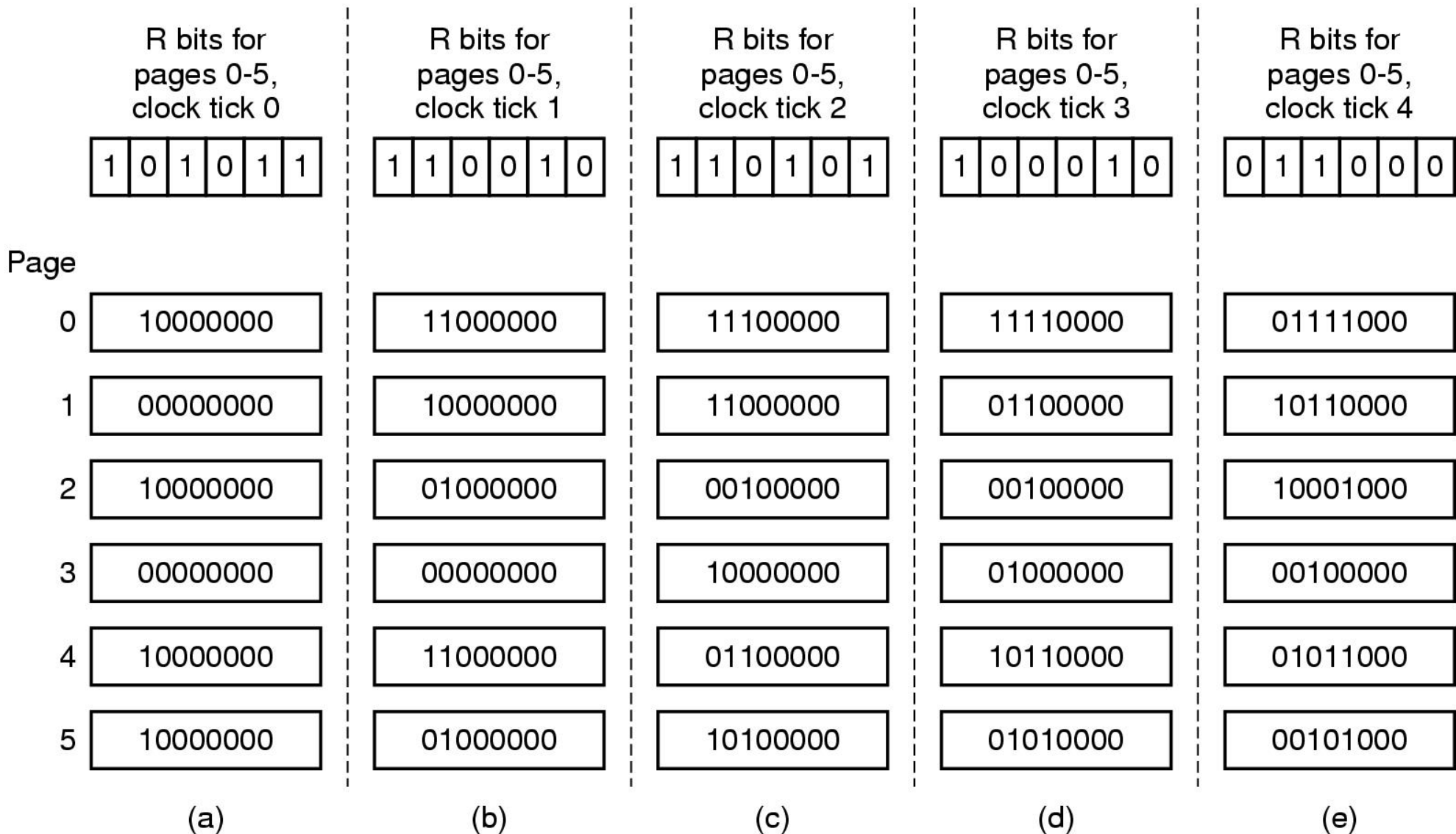
Yazılım ile LRU Benzetimi

Herhangi bir sayfa hatasında en küçük sayaca sahip olan sayfa çıkarılır.

Yazılımsal sayacın güncellenmesi işleminde, ilk önce sayaç 1 sağa kaydırılır. İkinci olarak da R biti o sayfanın en soluna eklenir.

Bu algoritma yaşlandırma(aging) algoritması olarak bilinmektedir.

Yazılım ile LRU Benzetimi



Yazılım ile LRU Benzetimi

Bit sayısı sınırlı olduğu için algoritma 9 saat atımı önce seçilmiş bir sayfa ile 1000 saat atımı önce seçilmiş olan sayfayı birbirinden ayıramaz. (sayacı 8 bit kabul edelim)

Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)

Sayfalamanın saf halinde süreçler bellekte hiçbir sayfaları olmadan çalışmaya başlarlar.

İşlemci ilk komutu (instruction) getirmeye çalıştığında bir sayfa hatası alınır.

Bu şekilde ilk komutun bulunduğu sayfanın belleğe getirilmiş olur.

Genel değişkenler ve yığın içinde çok kısa süre sayfa hataları oluşur ve bu sayfalarda belleğe getirilmiş olur.

Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)

Bu şekilde birkaç sayfa hatası ile süreç ihtiyaç duyduğu sayfaların belleğe yerleştirilmesini sağlar.

Bu taktiğe talep sayfalama (demand paging) denilir. Sayfalar daha önceden değil, istenildiklerinde yüklenirler.

Süreçler çalışma anında, sayfalarından küçük bir yüzdesine başvuru yapar. Buna başvuru yerelliği (locality of reference) denilmektedir.

Sürecin mevcut anda kullandığı sayıların kümesine **sayfa kümesi (working set)** denilmektedir.

Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)

Bir sürecin tüm çalışma kümesi bellekte olursa, süreç sayfalama hatası oluşturmada çalışır.

Bu durum süreç çalıştığı sayfaları değiştirene kadar yani çalışma kümesini değiştirene kadar bu şekilde devam eder.

Her sürecin çalışma kümelerinin izlenmesi ve çalışmadan önce bellekte olmasını sağlanmaya çalışılır. Bu modele çalışma kümesi (working set) denilir. Bu şekilde sayfa hatası oranı azaltılmaya çalışılır.

Süreç çalıştırmadan sayfanın yüklenemesine **ön sayfalama (prepaging)** denilir.

Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)

Çalışma kümesi algoritmasının gerçekleştirilmesi işlemi şu şekilde yapılır:

- Mevcut Sanal Zaman (Current Virtual Time)** : Bir sürecin çalışmaya başladığı andan itibaren kullanmış olduğu işlemci süresine denilir.

- Bir sürecin çalışma kümesi; belirlenmiş olan ***T*** sanal süre içerisinde erişilmiş olanlardır.

Yani süreç çalışmaya başladıktan sonra ***T*** süresi içerisinde erişim yapılanlar sürecin çalışma kümesini oluşturmaktadır.

Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)

Sayfaların yer değiştirmesi sürecin çalışma kümesinde olmayan sayfaların bulunması ve çıkarılması şeklinde yapılır.

R ve M bitleri her zamanki gibi değiştirilmektedir. Her saat atımında $R=0$ yapılır.

Oluşan her sayfa hatasında, sayfa tablosu çıkarmak için uygun bir sayfa bulmak amacıyla taranır.

R biti kontrol edilir.

Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)

Eğer $R=1$ ise, mevcut sanal zaman o sayfanın son kullanılma zamanına yazılır. $R=1$ olması sayfanın kullanıldığını gösterir.

Eğer $R=0$ ise, bu sayfa çıkarılmak için bir adaydır. Çıkartılıp çıkartılmayacağını belirlemek için, (Mevcut sanal zaman-bu sayfanın son erişilme süresi) hesaplanır ve bu değer **T** ile karşılaştırılır.

Eğer **$değer > T$** ise sayfa çalışma kümesinde değildir ve çıkartılır. Yeni sayfa buraya yüklenir. İşlem devam eder.

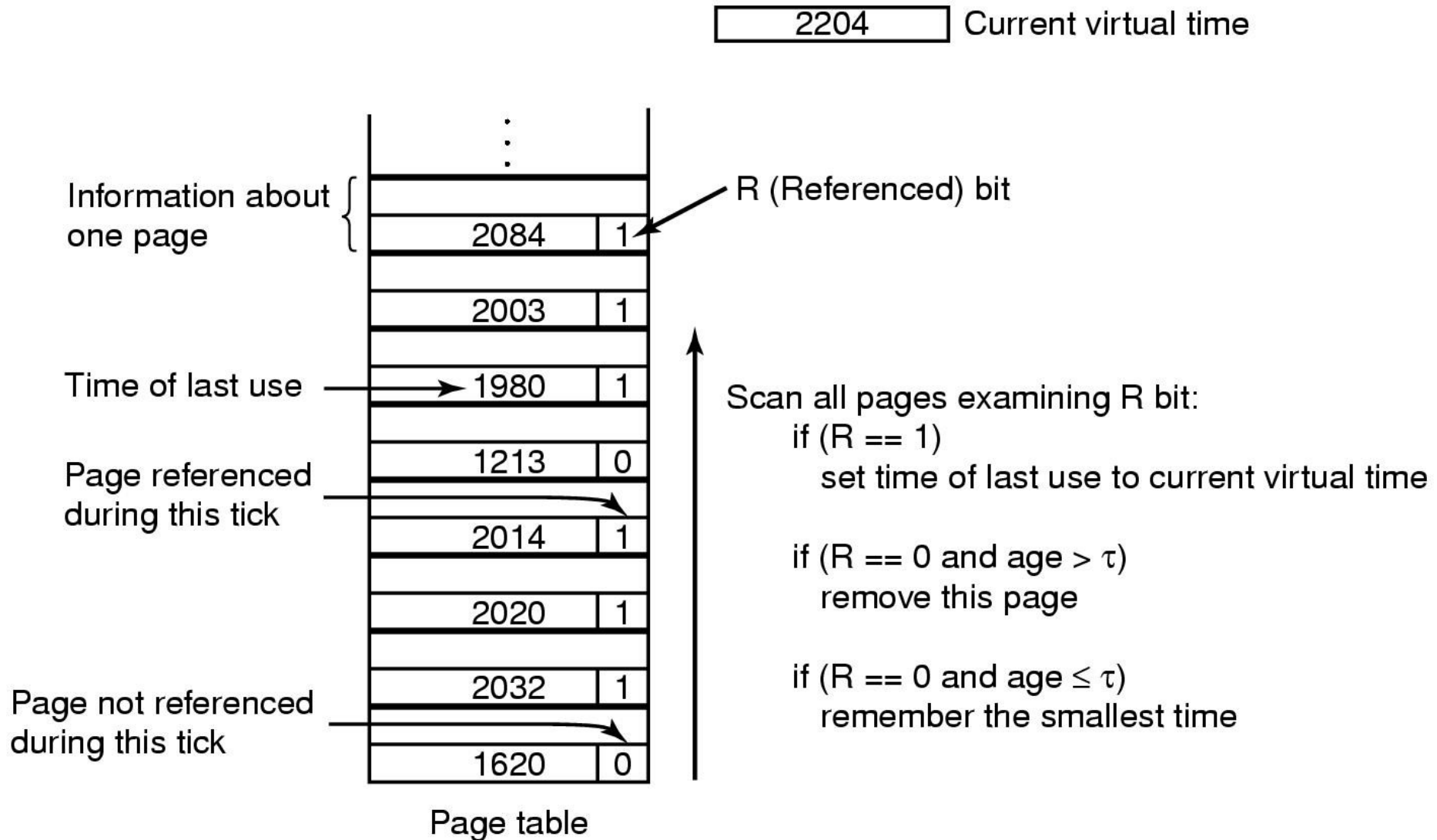
Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)

Eğer ***değer*** $\leq T$ ise bu sayfanın son erişilme zamanı kaydedilir.

Hiç çıkartılacak bir sayfa yok ise kayıt edilen sayfalardan en uzun süredir erişilmeyen çıkarılır.

Tüm sayfaların $R=1$ ise, rastgele bir tanesi çıkarılır.

Çalışma Kümesi Sayfa Yerdeğiştirme Algoritması (Working Set Page Replacement Algorithm)



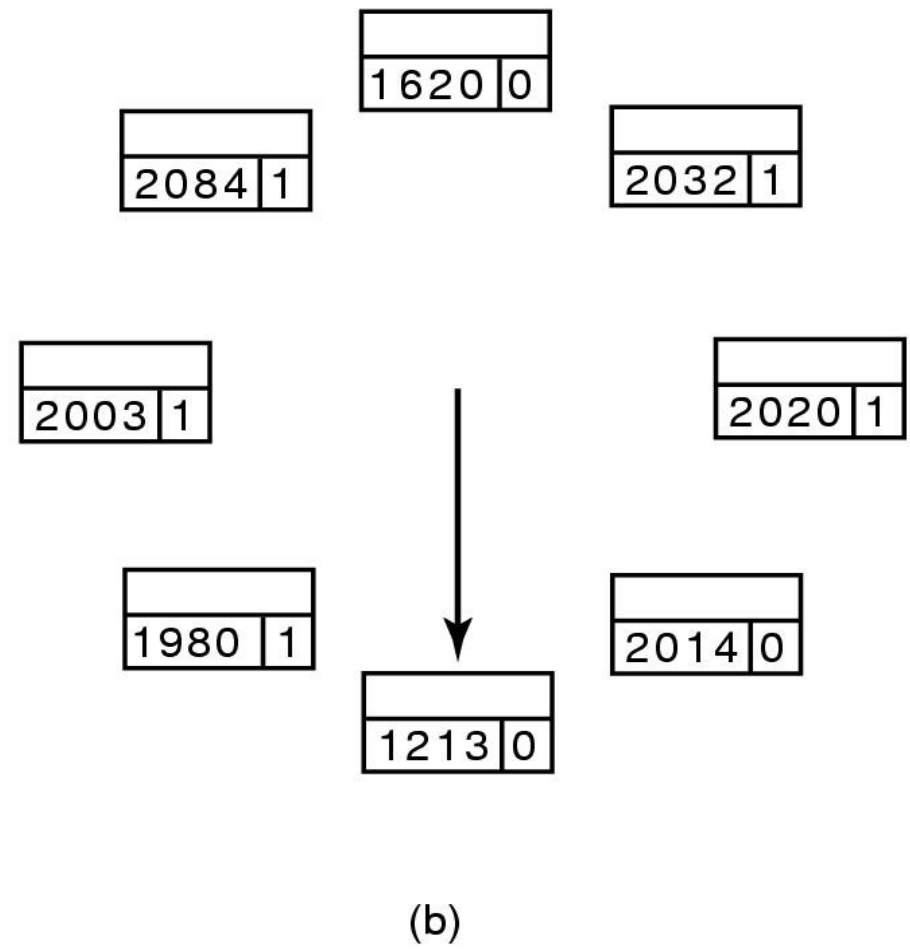
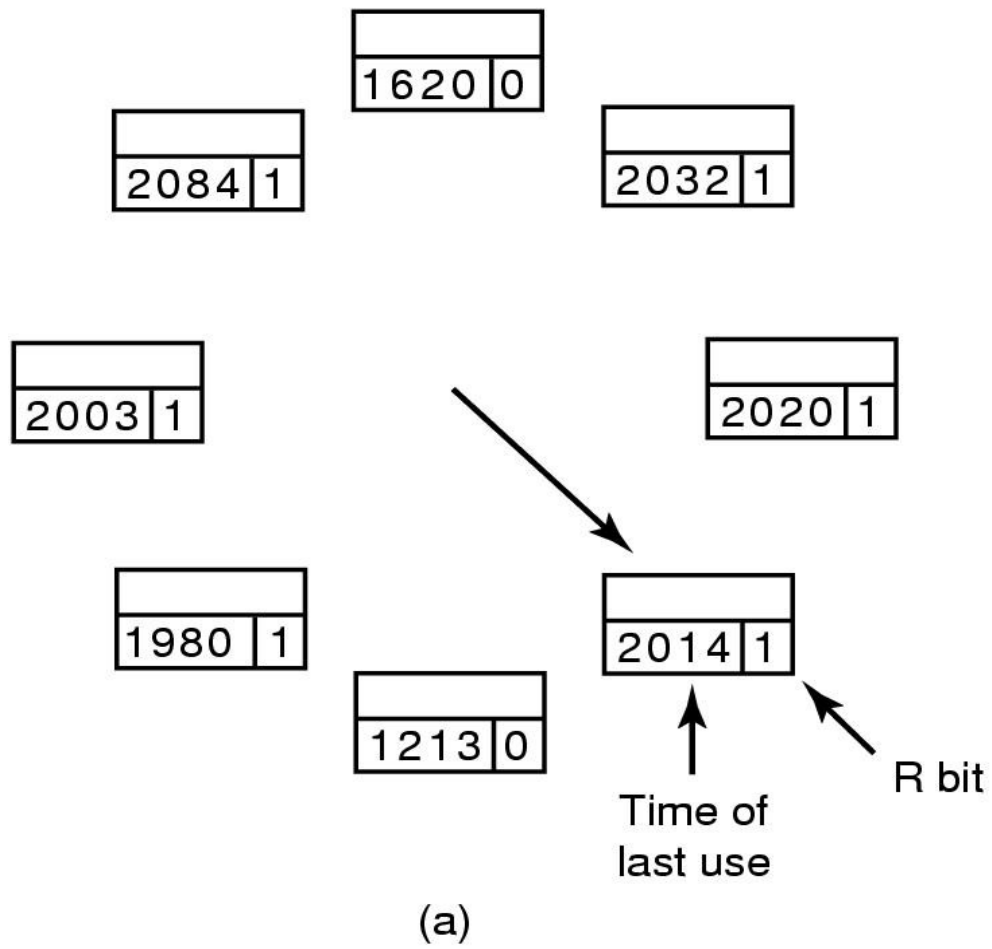
WSClock Yerdeğiştirme Algoritması (Working Set Clock page replacement algorithm)

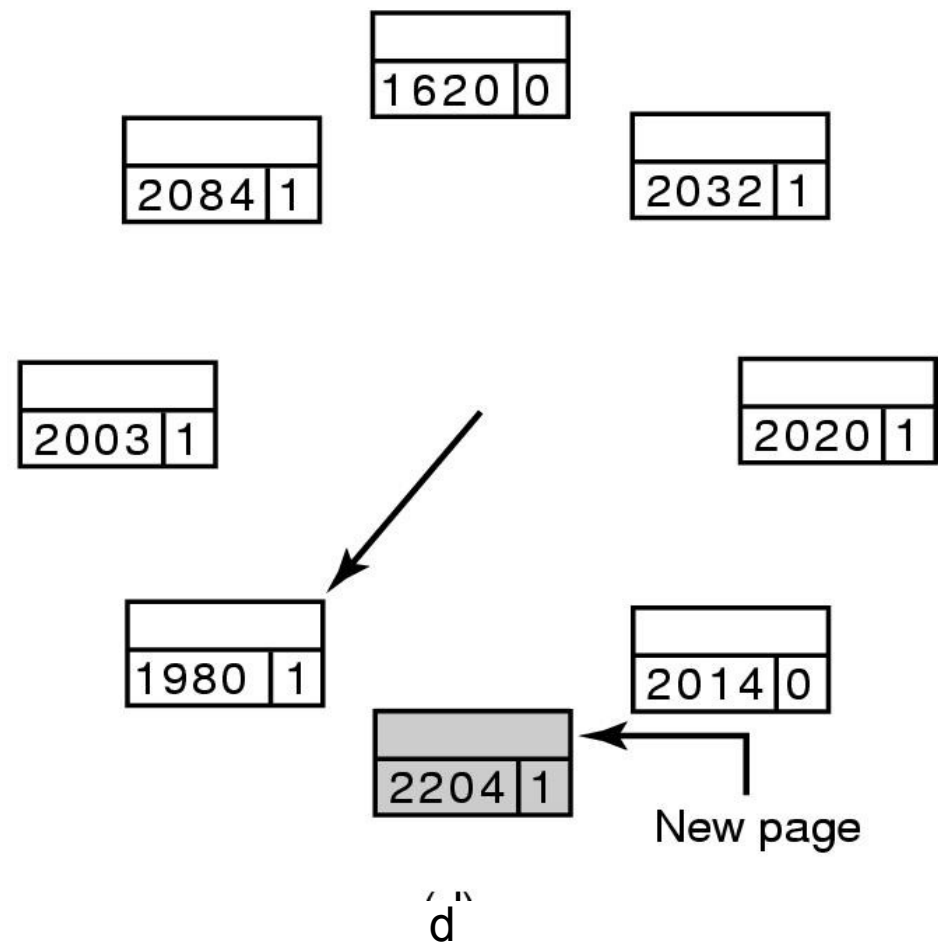
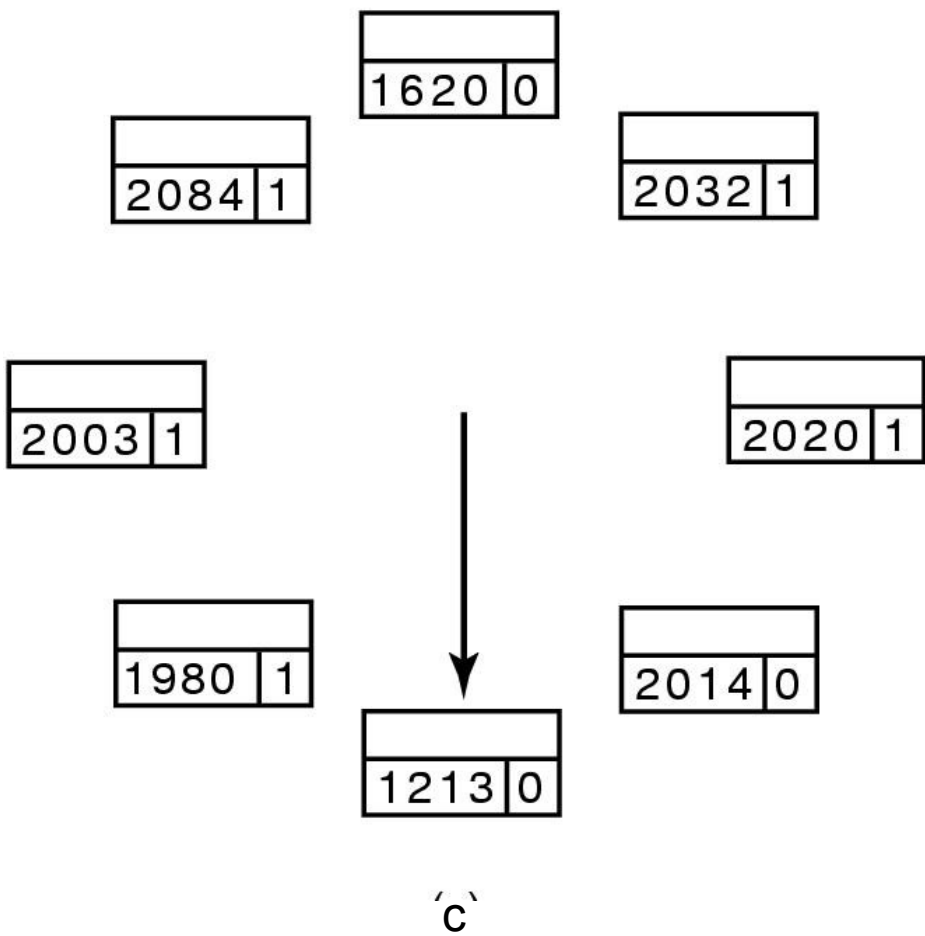
Temel çalışma kümesi algoritması, tüm sayfaların taranmasını gerektirdiği için kullanışsızdır.

Bu algoritma ile saat algoritması birleştirilerek WSClock algoritması oluşturulmuştur. Pratikte bu algoritma çok kullanılmaktadır.

Sayfa çerçevelerinin dairesel biçimde tutulduğu veri yapısı kullanılır. Başlangıçta bu liste boştur. İlk sayfa yüklenince listeye eklenir. Her sayfa eklenince bir dairesel yapı oluşur.

Her girdinin son kullanılma zamanı, R ve M bitleri ve diğer bilgileri vardır.





WSClock Yerdeğiştirme Algoritması (Working Set Clock page replacement algorithm)

Her sayfa hatasında göstergenin gösterdiği girdi ilk önce incelenir.

Eğer $R=1$ ise, aday olmaz ve $R=0$ yapılır. Gösterge bir sonraki kayıta ilerletilir. Bu kayıt için inceleme yapılır. Bu şekilde devam edilir.

Eğer $R=0$ ve (mevcut sanal zaman-son erişim zamanı) $> T$ ise, bu sayfanın kirli(dirty) olup olmadığına (verileri değiştirilmiş fakat diskteki kopyası eski ise) bakılır. Sayfa kirli ise diskteki kopyasının güncellenmesi için emir verilir ve gösterge bir sonraki kayıta ilerletilir.

Bu şekilde $R=0$ ve temiz kayıt aranır, bulununca da çıkarılır.

İşletim Sistemlerine Giriş

Bellek Yönetimi (Memory Management)