

Ödevi Sisteme Yükleme için Son Tarih = 08/05/2023 Pazartesi 23:00

Geç gönderimler ve ilgili kurallara uymayan gönderimler KESİNLİKLE kabul edilmeyecektir! Tüm kuralları dikkatli bir şekilde okuyunuz.

Diğer Kurallar:

- Her öğrenci ödevi kendisi yapmalıdır. Birlikte yapılan ödevler veya internetten alınan hazır kodlar (dosya okuma ve Queue interface kısmı hariç) **kopya** olarak değerlendirilecektir. Kontrol sonucu **kopya** tespit edilen ödevlere ödev notu olarak 0 verilecektir. Dersin öğretim elemanları kopya ödev gönderen öğrenciler hakkında **disiplin sürecini** başlatma hakkını saklı tutar.
- Dersin öğretim elemanları ödevi gönderen öğrencileri çağırıp (veya zoom üzerinden) çözümleri hakkında soru sorma ve anlattırma hakkını saklı tutar.
- Ödevler, UBYS (**Eders Değil!**) BSM304 İşletim Sistemleri sayfasındaki Ödevler sekmesinden sisteme yüklenmelidir. Mail yolu ile gönderilen ödevler kabul edilmeyecektir ve değerlendirmeye alınmaz.
- Uygulamanızın çalışıp çalışmadığı ve uygulamanızdaki eksik kısımlar vb. hakkında bilgi veren 1-2 satırlık kısa bir açıklama içeren .txt uzantılı metin dosyası (**açıklama.txt**) oluşturup, bunu proje klasörü ile birlikte sisteme yüklemeniz gerekmektedir.
- Proje klasörünü (açıklama içeren “açıklama.txt” dosyasıyla birlikte) sıkıştırıp (*.zip) uzantılı tek bir dosya sisteme yükleyiniz. ((*.rar) uzantılı dosyaların sisteme yüklenmesinde sıkıntı yaşanmaktadır ve ödeviniz değerlendirmeye alınmaz!)
- Proje klasörü yerine, sisteme tek bir class dosyası vb. farklı şekillerde yükleme yapanların ödevleri değerlendirilmeyecek ve 0 ile notlandırılacaktır.
- Projenizi “_ÖğrenciNo_Ad_Soyad” (örn. _2013510001_Ali_Bilir) şeklinde isimlendiriniz. Ayrıca, projede tanımladığınız her bir sınıfın ismi de öğrenci numaranız ile başlamalıdır (örn. _2013510001_PCB gibi).
- Kodunuza KESİNLİKLE **yorum satırları eklemeyiniz**.
- Son teslim tarihi ve saatine kadar ödevini sisteme yüklemeyenlerin ödevleri değerlendirmeye alınmayacaktır ve 0 ile notlandırılacaktır.

Başarılar.

Ödev Açıklaması:

Bu ödevde, Eclipse ve Java dilini kullanarak, işletim sisteminin PCB yaratma, prosesleri çalıştırma, prosesleri askıya alma ve kuyruğa yerleştirme, proses durumunu değiştirme vb. süreçlerini simüle eden ve aşağıda belirtilen işlemleri yapacak **dinamik bir konsol uygulaması** geliştirmeniz beklenmektedir. Diğer hususlar şu şekildedir:

- Uygulama çalıştığında ilk olarak “girdi.txt” adındaki dosyayı okumalıdır. “girdi.txt” dosyasında iki kısım bulunmaktadır. İlk kısım olan uygulamalar kısmında, uygulamalara ilişkin bilgiler (uygulama ismi, uygulamadaki komut sayısı, uygulamadaki I/O komutlarının yerleri ve türleri) yer almaktadır. İkinci kısım olan olaylar kısmında ise dış kaynaklı gerçekleşen olaylar ve olayların gerçekleşme zamanı yer almaktadır. Dış kaynaklı gerçekleşen tek olay run olayıdır. Run, bir programın çalıştırılması demektir. Ödevle birlikte örnek bir dosya sisteme yüklenecektir. Kendinizin de formata uygun farklı dosyalar oluşturup uygulamanızı test etmenizde yarar vardır.

- Örnek bir dosyanın ekran görüntüsü aşağıda yer almaktadır. Dosyadaki her bir değer arasında bir boşluk bulunmaktadır. Dosyadaki uygulamalar kısmının satır formatı şu şekildedir:

Uygulama_ismi uygulamanın_komut_sayısı I/O_komut_türü I/O_komutunun_sırası

Örnek: A.exe 20 ekran 8 ethernet 17 (Bu örnekte A.exe isimli uygulama 20 komuttan oluşmaktadır. 8. Komut ekran isteğinde bulunan, 17. Komut ise ethernet isteğinde bulunan I/O komutlarıdır. Benzer şekilde, başka I/O komutları varsa, onlar da yan yana yazılır. Herhangi bir I/O komutu içermeyen uygulamalar olabilir)

Herhangi bir I/O isteğine donanımın cevap verme süresi 3 saniyedir. Bu 3 saniye boyunca ilgili proses ilgili bekleme kuyruğunda olmalıdır.

Dosyadaki olaylar kısmının satır formatı şu şekildedir:

Run olayı için: run uygulama_ismi gerçekleşme_zamanı (Örnek: run A.exe 1) Bu örnekte, A.exe isimli uygulama 1. Saniyede çalıştırılmaktadır.

3. Çalıştırılan (run edilen) her bir uygulama için bir PCB (Process Control Block) yaratılmalıdır. PCB şu bilgileri içermelidir:
 - Proses numarası (pozitif bir tamsayıdır ve eşsiz olmalıdır.)
 - Proses durumu (5 durumdan bir tanesi)
 - Program sayacı değeri (bir sonraki adımda uygulamanın kaçınıcı komutu yürütülecek)
 - Hesap bilgileri (kullanılan CPU miktarı ve prosesin yaratılmasından itibaren geçen süre saniye olarak tutulmalıdır)
4. CPU'da (bilgisayarda) her saniyede 1 komut çalıştırıldığı varsayılacaktır.
5. Her bir uygulamanın (prosesin) time slice süresi 5 saniyedir. Eğer proses time slice süresinde sonlanmamış ise hazır kuyruğuna alınmalı ve hazır kuyruğundaki sıradaki proses CPU'ya atanmalıdır. (Eğer hazır kuyruğunda en az 1 proses varsa CPU hiç boş kalmamalıdır.)
6. İşletim sisteminin kendi maliyeti göz ardı edilecektir. Yani, işletim sisteminin CPU'yu hiç kullanmadığı varsayılacaktır.
7. Tüm proseslerin öncelikleri aynı kabul edilecektir.
8. Sistemde toplam 4 kuyruk vardır. Bunlar, hazır kuyruğu, ekran kuyruğu, disk kuyruğu ve ethernet kuyruğudur. Kuyruk için java dilinin kendi hazır Queue interface'ini veya benzer bir yaklaşımı kullanabilirsiniz.
9. Dersimizin 3. Hafta sunumu sayfa 8'de bulunan resme uygun olarak, bir olay gerçekleştiğinde prosesin durumu güncellenmelidir. Bir proses sonlandığında, o prosese ilişkin PCB de silinmelidir.
10. Programınız çalıştığında kullanıcıya sistemin durumunu görmek istediği saniyeyi soracaktır. Kullanıcı saniyeyi girdikten sonra, program o saniyedeki sistemin durumu hakkındaki bilgileri ekrana çıktı olarak verecektir. Daha sonra, programınız kullanıcıya ilgili saniyedeki PCB'sini görüntülemek istediği proses ismini soracaktır ve ismi girilen prosesin ilgili saniyedeki PCB bilgilerini ekrana çıktı olarak verecektir. (Örnek bir ekran görüntüsü aşağıda verilmiştir. Sizin uygulamanızın da aynı formatta ekrana çıktı üretmesi gerekmektedir. Formatı ve ekrandaki mesajların sırasını değiştirmeyiniz.
11. Projenizi yaparken nesneye yönelik programlama tekniğini göz önünde bulundurunuz. Dolayısıyla, projenizde belirli sayıda ve uygun şekilde sınıflar ve metotlar tanımlanmalıdır.

Örnek bir ekran görüntüsü aşağıda verilmiştir:

girdi.txt dosyası okundu.

Lütfen sistemin durumunu görmek istediğiniz saniyeyi giriniz.

20

CPU'da çalışan proses: B.com

Ready kuyruğu: hesapla.exe

Ekran kuyruğu: A.exe

Disk kuyruğu:

Ethernet kuyruğu:

PCB'si bulunan Prosesler: hesapla.exe A.exe B.com

20. saniyedeki PCB'sini görüntülemek istediğiniz proses ismini giriniz:

A.exe

A.exe isimli prosesin 20. Saniyedeki PCB bilgileri şu şekildedir:

Proses numarası: 1000

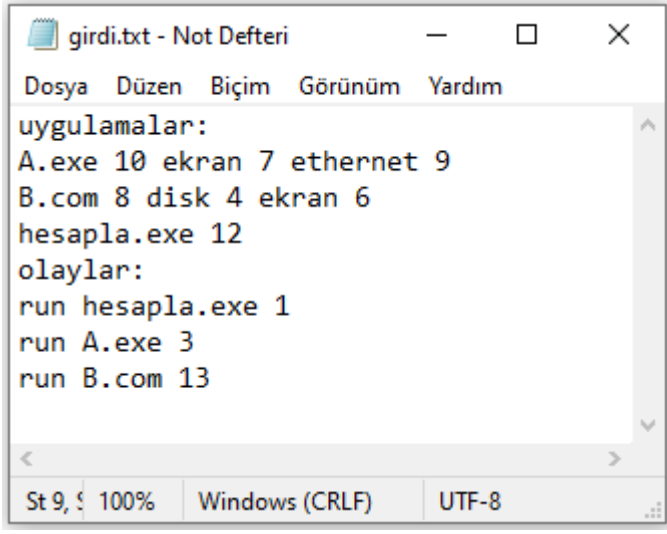
Proses durumu: waiting

Program sayacı: 8

Kullanılan CPU miktarı: 7 saniye

Prosesin yaratılmasından itibaren geçen süre: 17 saniye

Örnek bir girdi.txt dosyasına ilişkin ekran görüntüsü aşağıda verilmiştir:



```
girdi.txt - Not Defteri
Dosya Düzen Biçim Görünüm Yardım
uygulamalar:
A.exe 10 ekran 7 ethernet 9
B.com 8 disk 4 ekran 6
hesapla.exe 12
olaylar:
run hesapla.exe 1
run A.exe 3
run B.com 13
St 9, 100% Windows (CRLF) UTF-8
```