

Öğrenci No		Bölüm	Bilgisayar Mühendisliği
Adı Soyadı		Yıl / Yarıyıl	2025-2026 ☐ Güz ☒ Bahar
İmza		Tarih / Saat / Süre	31/03/2026 10.00 60 dakika
Dersin Kodu / Adı	BILM-300 İŞLETİM SİSTEMLERİ	Sınav Türü	☒ Ara Sınav ☐ Final ☐ Bütünleme
Dersin Sorumlusu	Dr. Öğr. Üyesi Sercan KÜLCÜ		☐ Mazeret ☐ Diğer:

Öğrenme Çıktıları	ÖÇ1	ÖÇ2	ÖÇ3	Toplam
Sorular	1-7	8-13	14-16	16 Soru
Puanlar	40	40	20	100 Puan
Alınan Puan				

Sınava İlişkin Açıklamalar

- Sınavın ilk 30 dakikasında salondan çıkmak yasaktır.
- Sınavda cep telefonu, akıllı saat vb. kullanmak; kalem, silgi alışverişi yapmak yasaktır.
- Cevaplarınız adım adım, anlaşılır ve düzenli olmalıdır. Soruların üzerine aksi belirtilmedikçe cevaplama yapmayınız. Anlaşılamayan düzensiz cevaplar ve soru üzerine verilen cevaplar puanlanmayacaktır.
- Sınav kurallarına mutlaka uyunuz. Sınav kurallarına uymayanlar hakkında disiplin soruşturması açılacaktır.

SORULAR

Soru 1: İşletim sisteminin temel görevlerini (en az 4 tane) açıklayınız. (5 puan)

İşletim sisteminin temel görevleri, donanım ile kullanıcı/uygulama arasındaki karmaşık etkileşimi yönetilebilir bir soyutlama düzeyine indirgemektir. Birincisi, süreç yönetimi, CPU zamanının adil ve verimli şekilde süreçlere dağıtılmasını sağlar. İkincisi, bellek yönetimi, fiziksel belleğin sanallaştırılması ve süreçler arasında güvenli paylaşımını gerçekleştirir. Üçüncüsü, dosya sistemi yönetimi, verilerin kalıcı depolama ortamlarında organize edilmesini ve erişimini düzenler. Dördüncüsü, giriş/çıkış (I/O) yönetimi, aygıt sürücülerini aracılığıyla donanım ile etkileşimi soyutlar. Bu görevlerin tümü, sistem performansı, güvenlik ve kullanılabilirlik arasında denge kurmayı amaçlar.

Soru 2: Önbellek (cache) kullanımı neden performansı artırır? İşlemci-bellek hiyerarşisi bağlamında açıklayınız. (5 puan)

Önbellek (cache), işlemci ile ana bellek arasındaki hız farkını azaltmak için kullanılan kritik bir ara katmandır. Modern işlemciler nanosaniye mertebesinde çalışırken, ana bellek erişimi daha yavaştır. Cache, zamansal (temporal) ve uzamsal (spatial) yerellik prensiplerinden yararlanarak sık erişilen verileri daha hızlı erişilebilir bir alanda tutar. Bu sayede ortalama bellek erişim süresi önemli ölçüde düşer. Bu mekanizma, bilgisayar mimarisinde “memory wall” problemine karşı geliştirilen en temel optimizasyonlardan biridir.

Soru 3: Sistem çağrısı (system call) ile normal fonksiyon çağrısı arasındaki farkı açıklayınız. (10 puan)

Sistem çağrısı (system call), kullanıcı modunda çalışan bir programın çekirdek modunda çalışan işletim sistemi hizmetlerine kontrollü erişimini sağlar. Normal fonksiyon çağrıları tamamen kullanıcı alanında gerçekleşirken, sistem çağrıları bir bağlam değişimi (mode switch) gerektirir ve donanım tarafından korunan çekirdek alanına geçiş yapılır. Bu durum güvenlik açısından kritik olup, kullanıcı programlarının doğrudan donanıma erişmesini engeller ve kontrollü bir arayüz sunar.

Soru 4: Kesme (interrupt) ve tuzak (trap) kavramlarını karşılaştırınız. (5 puan)

Kesme (interrupt) ve tuzak (trap), CPU'nun yürütme akışını değiştiren mekanizmalardır ancak kaynakları farklıdır. Interrupt, genellikle donanım tarafından tetiklenen ve asenkron gerçekleşen bir

olaydır (örneğin diskten veri gelmesi). Trap ise yazılım tarafından bilinçli olarak oluşturulan senkron bir olaydır (örneğin sistem çağrısı). Gerçek hayatta interrupt, kapının çalınması gibi dışsal bir uyarıyı temsil ederken; trap, kişinin kendisinin yardım istemesi gibi içsel bir tetiklemeyi temsil eder.

Soru 5: Monolitik çekirdek ve mikroçekirdek mimarilerini performans ve güvenlik açısından karşılaştırınız. (5 puan)

Monolitik çekirdek mimarisi, tüm sistem bileşenlerini (sürücüler, dosya sistemi vb.) tek bir çekirdek alanında barındırır. Bu yaklaşım yüksek performans sağlar çünkü bileşenler arası iletişim doğrudan yapılıdır. Mikroçekirdek mimarisi ise minimum çekirdek fonksiyonu içerir ve diğer hizmetleri kullanıcı alanına taşır. Bu yapı daha güvenli ve modülerdir ancak IPC maliyeti nedeniyle performans dezavantajı yaratabilir. Bu iki yaklaşım, performans-güvenlik dengesi açısından farklı tasarım felsefelerini temsil eder.

Soru 6: Aşağıdakilerden hangisi DMA'nın avantajlarından biridir? (5 puan)

- a) CPU kullanımını artırır b) Veri transferini yavaşlatır
c) CPU yükünü azaltır d) Bellek boyutunu artırır

Soru 7: Aşağıdakilerden hangisi süreçler arası iletişim (IPC) yöntemidir? (5 puan)

- a) Fork b) Exec c) Shared Memory d) Scheduler

Soru 8: Süreç (process) ve iş parçacığı (thread) arasındaki farkları açıklayınız. (Adres alanı, kaynak paylaşımı ve performans açısından değerlendiriniz.) (10 puan)

Süreç (process), kendi adres alanına sahip bağımsız bir yürütme birimidir; iş parçacığı (thread) ise bir süreç içerisindeki daha hafif yürütme birimidir. Süreçler birbirinden izole edilmiştir ve iletişimleri maliyetlidir (IPC gerekir). Thread'ler ise aynı adres alanını paylaşır, bu da daha düşük bağlam değiştirme maliyeti ve daha yüksek performans sağlar. Ancak bu paylaşım, veri yarışları ve senkronizasyon problemlerini beraberinde getirir. Bu nedenle thread kullanımı performans ile karmaşıklık arasında bir denge gerektirir.

Soru 9: Eşzamanlılık (concurrency) ve paralellik (parallelism) kavramlarını açıklayınız. Aralarındaki fark nedir? (10 puan)

Eşzamanlılık (concurrency), birden fazla işin aynı zaman diliminde ilerliyormuş gibi yürütülmesidir; paralellik (parallelism) ise bu işlerin fiziksel olarak aynı anda yürütülmesidir. Tek çekirdekli sistemlerde concurrency zaman paylaşımı ile sağlanırken, çok çekirdekli sistemlerde parallelism mümkündür. Bu ayrım, yazılım tasarımında kritik olup, concurrency daha çok programlama modeli iken, parallelism donanım kapasitesi ile ilişkilidir.

Soru 10: Bir sistemde Round Robin algoritması kullanılmaktadır ve zaman dilimi (time quantum) çok küçük seçilmiştir. Bu durumun sistem performansı ve kullanıcı deneyimi üzerindeki etkilerini açıklayınız. (5 puan)

Round Robin algoritmasında zaman diliminin çok küçük seçilmesi, sistemin aşırı sık bağlam değiştirmesine neden olur. Her context switch, CPU'nun mevcut sürecin durumunu kaydedip yenisini yüklemesini gerektirir ve bu işlem doğrudan faydalı iş üretmez. Bu nedenle sistemin toplam verimliliği düşer. Ancak kullanıcı perspektifinden sistem daha "duyarlı" hissedilebilir. Bu durum, performans ile etkileşimli yanıt süresi arasında klasik bir optimizasyon problemidir.

Soru 11: Bir süreç diskten veri okumak istediğinde hangi duruma geçer? (5 puan)

Bir süreç diskten veri okumak istediğinde CPU'dan ayrılarak blocked (engellenmiş) durumuna geçer. Bu süreç, I/O işlemi tamamlanana kadar CPU kullanmaz. İşlem tamamlandığında süreç tekrar ready durumuna alınır ve uygun olduğunda CPU'ya atanır. Bu durum geçişleri, işletim sisteminin verimli CPU kullanımını sağlamak için geliştirdiği temel mekanizmalardandır.

Soru 12: Bir süreç CPU'da aktif olarak çalışıyorsa hangi durumdadır? (5 puan)

- a) Hazır (Ready) b) Bekliyor (Blocked) **c) Çalışıyor (Running)** d) Sonlandırıldı (Terminated)

Soru 13: En kısa kalan süreyi temel alan zamanlama algoritması hangisidir? (5 puan)

- a) FCFS **b) SRTF** c) Round Robin(Sıralı) d) Priority Scheduling (Öncelik)

Soru 14: Kritik bölge (critical section) nedir? Neden önemlidir? Bir yarış durumu (race condition) örneği veriniz. (10 puan)

Kritik bölge, birden fazla süreç veya thread tarafından paylaşılan veriye erişilen kod bölümüdür. Bu bölgenin eşzamanlı erişimi kontrol edilmezse race condition (yarış durumu) ortaya çıkar. Örneğin iki thread aynı değişkeni aynı anda artırmaya çalışırsa sonuç beklenenden farklı olabilir. Bu nedenle kritik bölgeler mutex, semafor gibi mekanizmalarla korunmalıdır. Bu kavram, tutarlılık ve veri bütünlüğünün temelini oluşturur.

Soru 15: Aşağıdakilerden hangisi meşgul bekleme (busy waiting) kullanır? (5 puan)

- a) Gözleyici (Monitor) b) Condition Variable **c) Döngüzel Kilit (Spinlock)** d) Message Passing

Soru 16: Öncelik ters çevirme (Priority inversion) hangi durumda ortaya çıkar? (5 puan)

- a) Yüksek öncelikli süreç CPU'yu ele geçirirse **b) Düşük öncelikli süreç, yüksek öncelikli süreci engellerse**
c) Süreçler senkronize edilmezse d) Bellek yetersizse