

ANKARA UNIVERSITY - DEPARTMENT OF COMPUTER ENGINEERING  
COM 334/BLM 334 Operating Systems/İşletim Sistemleri - Final Examination/ Final Sınavı

1 June 2015 - Duration 90 minutes (Süre 90 dk)

Student Name	
Student Number	

1. Short Answers (%10)

- (a) Describe Belady's anomaly in page replacement (Tr: Belady anomalisi nedir? Açıklayınız?)  
(b) Segmentation serves a different purpose than paging. Describe the difference. (Segmentasyon sayfalardan farklı amaçlara hizmet eder. Farkı açıklayınız.)

1

Segmentasyon

Belady'nin bellek değişimi yöntemi bellek yönetimi yöntemidir. Segman bir de persan (segment) kavramından oluşur. Bir programın bellek ihtiyacı, prosedür, veri ve global değişkenler, yığın, pilet bellekleri ve benzerleri için ayrıdır. Her segmentin kendi bir adresi vardır. Adresler de segmentin başlangıcı ve sonu olarak belirtilir. Bu şekilde adresler de segmentin başlangıcı ve sonu olarak belirtilir.

Segmentasyon

Segmentasyon yöntemi bellek yönetimi yöntemidir. Segman bir de persan (segment) kavramından oluşur. Bir programın bellek ihtiyacı, prosedür, veri ve global değişkenler, yığın, pilet bellekleri ve benzerleri için ayrıdır. Her segmentin kendi bir adresi vardır. Adresler de segmentin başlangıcı ve sonu olarak belirtilir. Bu şekilde adresler de segmentin başlangıcı ve sonu olarak belirtilir.

\* Segmentasyon yöntemi bellek yönetimi yöntemidir.

Segmentasyon yöntemi bellek yönetimi yöntemidir. Segman bir de persan (segment) kavramından oluşur. Bir programın bellek ihtiyacı, prosedür, veri ve global değişkenler, yığın, pilet bellekleri ve benzerleri için ayrıdır. Her segmentin kendi bir adresi vardır. Adresler de segmentin başlangıcı ve sonu olarak belirtilir. Bu şekilde adresler de segmentin başlangıcı ve sonu olarak belirtilir.

Segmentasyon

Bellek yönetimi yöntemi bellek yönetimi yöntemidir. Segman bir de persan (segment) kavramından oluşur. Bir programın bellek ihtiyacı, prosedür, veri ve global değişkenler, yığın, pilet bellekleri ve benzerleri için ayrıdır. Her segmentin kendi bir adresi vardır. Adresler de segmentin başlangıcı ve sonu olarak belirtilir. Bu şekilde adresler de segmentin başlangıcı ve sonu olarak belirtilir.

2. (%10) Draw a graph showing the child/parent relationship of processes created by the following code fragment. (Aşağıda verilen kod ile yarattılan süreçler arasındaki ilişkiyi gösteren bir graf çiz)

pid = fork();

if (pid != 0)

3. (5%10) Consider the following set of processes with arrival times and CPU execution times given in milliseconds. (Aşağıdaki geliş zamanı (arrival) ve çalışma süresi (length) verilen prosesler için a) kesilen (ik gelen öncelikli b) kesilmeyen SRJF.)

Process	Arrival	Length
P <sub>1</sub>	0	3
P <sub>2</sub>	1	6
P <sub>3</sub>	2	2
P <sub>4</sub>	6	3
P <sub>5</sub>	10	1

Draw and label Gantt charts illustrating the execution of these processes using:

- immediately preemptive FCFS
- preemptive SRJF scheduling

4. (15%) Consider the following proposed solution to the Bounded Buffer producer/consumer problem. The semaphores are initialized to full = 0, empty = n, and mutex = 1.

a) Is there a problem with this solution? Why?

b) What effect would reversing the order of the P operations in lines 3 and 4 for the consumer have? Explain.

(Tr: Sınırlı tampon problem için önerilen kodda semaforlar full = 0, empty = n, and mutex = 1 olarak atanmıştır. a) Bu çözüm yeterlidir? Neden? b) Satır 3 ve 4 deki P operasyonlarının yerleri değiştiril olur? Açıklayınız.)

Producer Process

```
do{
// produce an item in nextp
P(empty);
P(mutex);
// copy from nextp to buffer
V(mutex);
} while(1);
```

Consumer Process

```
do{
P(full);
P(mutex);
// copy an item from buffer to nextc
V(mutex);
// consume the item in nextc
} while(1);
```

Line  
Lin  
Lin  
Line  
L  
Li

ANKARA UNIVERSITY - DEPARTMENT OF COMPUTER ENGINEERING  
COM 334/BLM 334 Operating Systems/İşletim Sistemleri - Final Examination/ Final Sınavı  
1 June 2015 - Duration 90 minutes (Süre 90 dk)

Student Name	TUTULU ERGEC
Student Number	102.8063.0

1. Short Answers (%10)

- (a) Describe Belady's anomaly in page replacement. (T) Belady anomalis nedir? Açıklayınız?  
(b) Segmentation serves a different purpose than paging. Describe the difference. (Segmentasyon sayfalamadan farklı amaçlara hizmet eder. Farkı açıklayınız.

2. (%10) Draw a graph showing the child/parent relationship of processes created by the following code fragment. (Aşağıda verilen kod ile yaratılan processler arasındaki ata/çocuk ilişkisini gösteren bir graf çiz

```
pid = fork();  
  
if (pid != 0)  
    fork();  
  
fork();
```



ANKARA UNIVERSITY - DEPARTMENT OF COMPUTER ENGINEERING  
COM 334/BLM 334 Operating Systems/İşletim Sistemleri - Final Examination/ Final Sınavı  
1 June 2015 - Duration 90 minutes (Süre 90 dk)

Student Name	
Student Number	

1. Short Answers (%10)

- (a) Describe Belady's anomaly in page replacement. (Tr: Belady anomalisi nedir? Açıklayınız?)  
(b) Segmentation serves a different purpose than paging. Describe the difference. (Segmentasyon sayfalamadan farklı amaçlara hizmet eder. Farkı açıklayınız.)

Segmentasyon Belleğin bellek birimlerini destekleyen bellek yönetme tekniğidir. Program bir tek parçanın (segment) birleşiminden oluşur. Ana programda bulunan fonksiyonlar, prosedürler, lokal ve global değişkenler, yığın, pilet bölümleri bu bölümlerde yer alır. Her segmentin belli bir adresi vardır. Her adres segmentin offset adreslerinden oluşur. Her offset adresler de segment numarası ve offsetten oluşur.

Sayfalamada Belleğin bellek birimlerini destekleyen bellek yönetme tekniğidir. Program bir tek parçanın (segment) birleşiminden oluşur. Ana programda bulunan fonksiyonlar, prosedürler, lokal ve global değişkenler, yığın, pilet bölümleri bu bölümlerde yer alır. Her segmentin belli bir adresi vardır. Her adres segmentin offset adreslerinden oluşur. Her offset adresler de segment numarası ve offsetten oluşur.

\* Sayfalamada her bellek birimi'si vardır. Segmentasyon ise her programın birimlerinden biri biri gerçekleştirilir. Sayfalamada bellekte işi bilen programın bir adresi tespit edip kullanılmayan sayfalar. Aynı bellek daha görünür ve kullanılır. İhtiyaç duyulan olan bölümlerden büyük bölümlerde = alanı daha görünür. Ve = den alınır. Bellek likli'lik olabilir. Çözümler.

2. (%10) Draw a graph showing the child/parent relationship of processes created by the following code fragment. (Aşağıda verilen kod ile yaratılan processler arasındaki ata/çocuk ilişkisini gösteren bir graf çiz)

```
pid = fork();
```

```
if (pid != 0)
```



