



**TOBB Ekonomi ve Teknoloji
Üniversitesi
Bilgisayar Mühendisliği Bölümü**

18 Aralık 2023
BİL 361 – Bilgisayar Mimarisi ve Organizasyonu
2023 – 2024 Öğretim Yılı
Güz Dönemi
Final

Adı	Soyadı	Öğrenci Numarası	Bölümü

Açıklamalar

- Sınavı çözmeye başlamadan önce tüm açıklamaları ve soruları okuyun. Sınavda toplam 4 soru var ve soruların toplam değeri 110 puandır. Bütün soruların değeri köşeli ayraç ile belirtilmiştir. Zamanınızı tek bir soru üzerinde harcamayın; tüm soruları yanıtlamaya çalışın. Sınav süresi 150 dakikadır.
- Tüm sınav kâğıtlarını geri vermeyi unutmayın. Adınızı tüm sayfalara yazın.
- Aşağıda verilen namus sözünü imzalayın (isteğe bağlı).
- Başkasının kâğıdına bakma ve başkasıyla konuşma kopya olarak değerlendirilecektir.
- Kitap, defter, kâğıt, hesap makinesi, cep telefonu, bilgisayar vb. araç gerecin kullanımı yasaktır.
- Sonucu yanlış olan yanıtlar puan alamayabilir. Gidiş yolunun ayrıntılı gösterilmesi sorudan puan alınması için gereklidir ancak yeterli değildir. **Açıklamasız işlem yapmamaya özen göstermeniz alacağınız puanı artıracaktır.**

Bu sınavda kimseye yardım etmedim, kimseden yardım almadım ve kimseyi yardım alırken ya da başkasına yardım ederken görmedim.

İmza:

Soru (Puan)	1 (15)	2 (30)	3 (30)	4 (35)	Toplam (110)
Not					

1. [15 Puan] Boşluk Doldurma

Şıklarda verilen boşlukları doldurun.

- a-) [2 Puan] Giriş-çıkış aygıtlarından bilgi almak için çokça kullanılan iki yöntem vardır. Bunlar ve
- b-) [1 Puan] Giriş-çıkış aygıtlarına yol atama önceliği yönteminde donanımda sabittir.
- c-) [2 Puan] Bellekten hassas verileri çalmaya yönelik güncel iki saldırı çeşidi ve
- d-) [2 Puan] Giriş-çıkış aygıtlarına yol ataması sırasında dikkat edilen iki faktör ve
- e-) [3 Puan] Giriş-çıkış aygıtları için yolun hızını belirleyen faktörler ve ve
- f-) [2 Puan] Giriş-çıkış aygıtlarında yolun iki ucu arasında farklı aktarım yöntemleri aktarım ve aktarımdır.
- g-) [1 Puan] İşlemciyi aradan çıkararak bellekten giriş-çıkış aygıtlarına direkt olarak veri iletilmesini sağlayan yapı olarak adlandırılır.
- h-) [2 Puan] Giriş-çıkış aygıtlarında yol kısıtları kısıtlar ve kısıtı olarak ikiye ayrılır.

2. [30 Puan] Önbellek Tutarlılığı

4 adet çekirdek bulunan, her çekirdeğin doğrudan eşlemeli 4096 bayt kendi önbelleği olduğu, veri öbeklerinin **1024 bayt** olduğu, **bayt adresleme** ve **MESI** protokolünün kullanıldığı 32 bitlik bir sistemde; başlangıçta önbellekler tabloda gösterilen durumdayken yine aşağıda gösterilen yükle(load) ve sakla(store) buyrukları ilgili çekirdeklerde (örn. Ç[0] demek Önbellek 0'a sahip çekirdek demek) sırasıyla çalıştırılıyor. Her buyruk bitiminde önbelleklerin durumunu adım adım çizerek gösterin.(**Tabloların değişen satırlarını** göstermeniz yeterli fakat kaçınıcı satır ve hangi önbellekte olduğunu açıkça gösterin.) Önbelleklerin son durumu için aşağıda boş olarak verilen tablonun tamamını doldurun. (M: Modified, E: Exclusive, S: Shared, I: Invalid)

Not: Önbelleklerin başlangıç durumunun tutarlı olup olmamasını önemsemeyin.

Önbellek 0		Önbellek 1	
Etiket	MESI Durumu	Etiket	MESI Durumu
0x555FE	E	0x12312	S
0x3210A	S	0x11111	I
0xCCCCC	S	0xCCCCC	S
0x66DE5	I	0x66DE5	M
Önbellek 2		Önbellek 3	
Etiket	MESI Durumu	Etiket	MESI Durumu
0x18102	M	0x18102	I
0xCCCCC	S	0xCCCCC	S
0x21721	E	0x12312	S
0x3210A	S	0x3210A	S

Çalıştırılan Çekirdek	Buyruk
Ç[1]	Sakla R1, 0x3210A007
Ç[0]	Yükle R2, 0x1810277B
Ç[3]	Yükle R3, 0x555FEAAA
Ç[0]	Sakla R4, 0xCCCCCFFF
Ç[2]	Yükle R5, 0xCCCCCFF8
Ç[3]	Sakla R6, 0xCCCCCFF7
Ç[0]	Yükle R7, 0xCCCCCFF0
Ç[3]	Yükle R8, 0x1810277B

Önbellek 0		Önbellek 1	
Etiket	MESI Durumu	Etiket	MESI Durumu
Önbellek 2		Önbellek 3	
Etiket	MESI Durumu	Etiket	MESI Durumu

3. [30 Puan] Sanal Bellek

Aşağıdaki tablolarda veri önbellegi, Etkin Sayfalar Önbellegi (ESÖ) durumu ve adres dönüşüm tablosu verilmiştir. Veri önbellekleri **2 yollu kümeli** ilişkilidir, yaz ve yerini ayır ile rastgele çıkar politikası kullanılmaktadır. Sistemde **bayt adresleme** kullanılmaktadır, tüm adresler 16 bittir ve **veri öbekleri 4 bayttır**. Veri tablolarındaki her karakter **1 bayt** boyutundadır ve adresleme öbeklerin en sağ karakteri 0. bayt olacak şekildedir. ESÖ **doğrudan eşlemelidir**.

Veri önbellegine gerçek sayfa numaralarıyla, ESÖ ile aynı anda erişilmektedir ve önbelleg yolları olabilecek en büyük boyutlardadır.

Veri Önbellegi Yol 1		Veri Önbellegi Yol 2		Etkin Sayfalar Önbellegi		Adres Dönüşüm Tablosu	
Etiket	Veri	Etiket	Veri	Etiket	Veri	Sanal	Gerçek
0x3FC	gizl	0x201	iyer	0xFF	0x2AA	0x80F	0x201
0x2AA	elif	0x102	nisa	0x3FC	0x102	0x2AA	0x1AA
0x526	batA	0xAA	070F	0xC3	0x3FC	0x72F	0x3FC
0x00A	şval	0x0AB	YCT0	0x155	0x201	0x1AA	0x4EF
						0x333	0x155
						0x155	0x2BF
						0x2AC	0x72F
						0x3FC	0x2AA

- [4 Puan]** Sanal sayfa boyutu kaçtır?
- [8 Puan]** Aşağıdaki buyruklar sırasıyla çalıştırılırsa yukarıdaki tablolarda ve yazmaçlarda neler değişir ?
li x2, 0x3FC4 # erişilecek adres
lw x1, 0(x2)
- [8 Puan]** Aşağıdaki buyruklar sırasıyla çalıştırılırsa yukarıdaki tablolarda neler değişir?
li x2, 0x1558 # erişilecek adres
li x1, oguz # 4 bayt veri
sw x1, 0(x2)
- [10 Puan]** Ekranı “gizliyer” bastırmak istiyoruz ve stringimiz çoktan önbellegte mevcut. Seçtiğiniz ardışık yazmaçlara (örn, x1 ve x2) bu veriyi içeren öbekleri okuyacak assembly kodu yazın.

li, xi, ANLIK bit kısıtı olmadan gelen tüm anlığı hedef yazmaca yazar (xi = ANLIK)
lw xi, 0(xj) buyruğu xi yazmacına xj adresindeki 32 biti okur
sw xi, 0(xj) buyruğu xj adresine xi'deki 32 biti yazar

...
**Tablonun sorudan
bağımsız kısımları
kırpılmıştır**

4. [35 Puan] Güncel İşlemciler

X Programı	Y Programı	Z Programı
li x1, 0 li x2, N li x3, 0 li x4, N döngü: add x6, x1, x3 add x7, x1, x4 lw x8, 0(x6) lw x9, 0(x7) addi x1, x1, 4 sub x8, x8, x9 add x9, x9, x8 sw x8, 0(x6) sw x9, 0(x7) blt x1, x2, <döngü>	li x1, 0 li x2, N li x3, 0 li x4, N döngü: add x6, x1, x3 add x7, x1, x4 lw x8, 0(x6) lw x9, 0(x7) addi x1, x1, 16 sub x8, x8, x9 add x9, x9, x8 sw x8, 0(x6) sw x9, 0(x7) blt x1, x2, <döngü>	li x1, 0 li x2, M li x3, 3N li x4, 0 li x5, 1 li x7, 0 döngü: add x6, x4, x5 add x4, x5, x0 addi x1, x1, 4 add x5, x6, x0 add x7, x7, x6 blt x1, x2, <döngü> sw x6, 0(x3) sw x7, 4(x3)

X ve Y programı arasında kalın yazılı addi buyruğu farklıdır.

Not: **li** buyruklarının **anlık** (örn., N, M) değerlerin büyüklüğünden bağımsız olarak yazmaca başarılı bir şekilde N yazdığını varsayın. Yani anlık bitleri yetersiz olduğu için herhangi bir veri kaybı olmayacak.

Sizlere yukarıdaki üç adet program veriliyor. Bu programlar özellikleri aşağıda verilen DUTA2 adlı işlemcide çalıştırılacaktır:

- Yazmaçlar **64 bit** genişliğindedir ve veri öbek boyutu **256 bayttır**.
- İşlemcide sıralı yürütüm yapılmaktadır ve **5 aşamalı çok yollu, çok iş parçacıklı boru hattı** vardır. Bunlar sırasıyla **Getir, Çöz, bağımlılık, yÜrüt** ve **geri Yaz** aşamalarıdır. Sistemde **64KB** boyutunda doğrudan eşlemeli, yaz ve yerini ayır politikası kullanan önbellek bulunmaktadır.
- **Getir** aşamasında 3 bağımsız program sayacı bulunmaktadır ve her çevrim toplam **2 ya da 3 buyruk** getirilebilmektedir (sorularda hangisinin geçerli olduğu olduğu belirtilmiştir).
 - o Getirilen buyruklar **farklı** programlardan birer buyruk ya da **aynı** programdan iki buyruk şeklinde olabilir.
 - o İşlemcide kahin öngörücü mevcuttur ve her zaman doğru sonuç tahmin edilir.
- **Çöz** aşamasında **her çevrim 3** buyruk çözülebilmektedir. Yazmaçlar bu aşamada okunur.
- **bağımlılık** aşamasında buyruklar arasındaki bağımlılıklar çözülür.
 - o Geçerli buyruklar eğer boşta işlem birimi varsa **yÜrüt** aşamasına iletilir ve çalıştırılır.
 - o Bir programın buyrukları arasında çözülemeyen bir bağımlılık bulunduğunda buyruklar kendi içlerinde duraklatılır. Bir programın buyrukları duraklatılırken diğer programların buyrukları boru hattında yeterli işlem birimi varsa ilerleyebilirler.
 - o Her çevrim 2 ya da 3 buyruk getirildiğinden ilki Atlar öngürülen dallanma sonrası getirilmiş ikinci ile varsa üçüncü buyruk gibi geçersiz buyruklar iptal edilir ve boru hattından atılır.
- **yÜrüt** aşamasında **3 adet AMB** ve **3 adet Bellek İşlem Birimi (BİB)** bulunmaktadır. Dallanmalar kendilerine özel bir birim içerisinde bu aşamada 1 çevrimde çözülür.
 - o AMB (li, add, sub, addi) tüm işlemlerini 1 çevrimde tamamlar.
 - o BİB bellek önbellekte bulunamayan okuma ve yazma (lw, sw) işlemlerini 400 çevrimde, önbellekte bulunan okuma ve yazma işlemlerini 1 çevrimde tamamlar.
 - o **yÜrüt** aşamasının sonundan başına veri yönlendirmesi vardır.
 - o **yÜrüt** aşamasında aynı anda farklı programların buyrukları işlenebilir.
- **geri Yaz** aşamasında tamamlanmış buyruklar mimari durumu doğru şekilde ve sırada güncelleyerek boru hattından ayrılırlar.
- İşlemci **4 GHz** saat vuruş sıklığına sahiptir. Yani her çevrim **0.25** nanosaniye sürmektedir.
- Önbelleğe getirilen öbekler, öbeği getiren programın erişimleri bitmeden çıkarılmamaktadır. Yani, öbeğe yapılan ilk erişim dışında ana bellek maliyetini değerlendirmenize gerek yok.

- a. **[15 Puan]** İşlemcide buyruk düzeyinde ince taneli (fine-grained) çok iş parçacığı çalıştırma kullanmak istiyoruz (her programdan 1 buyruk getirilecek). Herhangi bir sebepten boru hattında duraklatılan ya da tamamlanan program yerine diğer programların buyrukları getirilmeyecektir.

$N = 128$ ve $M = 256$ için üç program da tamamlandığında toplam çalışma süresi kaç nanosaniyedir? İşlemlerinizi gösterin ve detaylıca açıklayın.

- b. **[20 Puan]** İşlemcide buyruk düzeyinde kaba taneli (coarse-grained) çok iş parçacığı çalıştırma kullanmak istiyoruz. Bunu da şu şekilde yapacağız.

- Her çevrim program sayaçlarından yalnızca bir tanesine ait 2 buyruk getirilebilir.
- İlk başta programlar **döngü etiketi öncesindeki tüm buyrukları tamamlanana kadar** (li buyrukları) çalıştırılacaktır. İşlemlerinizi yaparken bu kısımlar yok gibi davranın.
- Her program için $[X_0, X_1, \dots, X_i]$, $[Y_0, Y_1, \dots, Y_i]$ ve $[Z_0, Z_1, \dots, Z_i]$ getirme çevrim sayısı X, Y, Z dizileri tanımlıdır. i, j ve k sayıları **en az 0 olabilir**.
- İşlemci program başladığında önce X_0 çevrim boyunca X programından, Y_0 çevrim boyunca Y programından ve Z_0 çevrim boyunca Z programından buyruk getirilir. Daha sonra X_1 çevrim ilk programdan buyruk getirilir ve örüntü bu şekilde devam eder.
- Dizi sonuna erişilirse getirme sırası o dizi için başa döner.
 $X_0 Y_0 Z_0, X_1 Y_1 Z_1, \dots, X_i Y_i Z_i, X_0 Y_{i+1} Z_{i+1}$
- Herhangi bir sebepten boru hattı duraklatılırsa programlar arası erken geçiş **yapılmaz** (soru sonundaki notları inceleyin). Programlardan birisi tamamlanırsa sürekli olarak kalan programın dizilerine göre buyrukları getirilir.

Program çalışma süresini en düşük yapacak X, Y ve Z dizilerini hesaplayınız. Hesapladığınız dizi için toplam çalışma süresini N ve M sayıları cinsinden ifade edin. İşlemlerinizi gösterin ve detaylıca açıklayın.

Notlar:

Dizinizin ilgili elemanı X programından 10 buyruk getirilmesini gerektiriyorken 4. getirilen buyruk sonrasında boru hattı duraklatılırsa, işlemci bu duraklama sonlanana ve kalan 6 buyruk da getirilene kadar aynı programda kalır.

Dizilerinizde tekrarlı kısımlar ve tanımlar kullanabilirsiniz.

Örn, $X = [100 \text{ tane } 10, 25, 27, 100 \text{ tane } 30]$

Analizinizi kolaylaştırmak adına N ve M sayıları hakkında aşırı kısıtlayıcı olmayan varsayımlarınız olabilir.

GEÇERLİ Örn, N sayısı 10'un tam katıdır ve M tek tir.

GEÇERSİZ Örn, N 25'tir ve M 64'tür.