	SÜLEYMAN DEMİREL ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ			TARİH
				30/05/2017
BIL-304 BİLGİSAYAR MİMARİSİ	ÖĞRENCİ ADI ve SOYADI	NO	İMZA	ALDIĞI PUAN/100

2016-2017 BAHAR DÖNEMİ FİNAL SINAVI SORULARI

Soru 1: (10 Puan) Kodlarımızı işaretli sayıların 2 tümleyen yöntemiyle ifade edildiği 8-bitlik bir makinada çalıştırdığımızı varsayarak aşağıdaki tabloda verilen boşlukları doldurunuz. `int` 8-bit ile ve `short int` 4-bit ile kodlanmaktadır. "—" içeren hücreleri doldurmanız istenmemektedir. Tabloda yer alan değişkenler için şu tanımlamaları kullanınız:

```
short int sy = -6;
int y = sy;
int x = -47;
unsigned int ux = x;
```

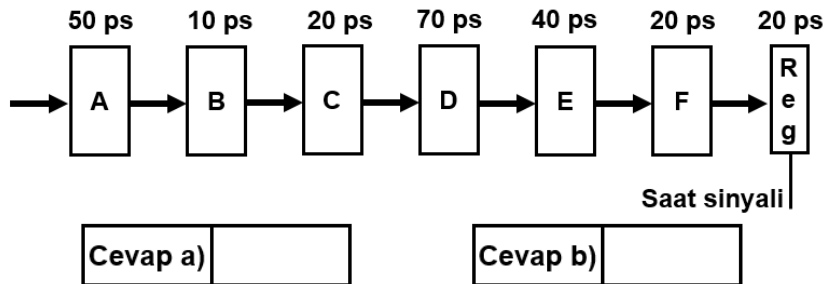
İfade	Onluk gösterimi	İkili gösterimi
-	-37	
-		0101 0010
ux		
y		
$x \gg 3$		
$T_{Max} + T_{Min}$		

Soru 2: (10 Puan) Bir işlemciye ait kombinyonel devre katmanı inceleniyor ve aşağıdaki şekilde verilen gecikmelere sahip, 6 blok elde ediliyor. Bu blokların daha alt bloklara ayrılmadığını ve sisteme eklenecek her bir pipeline kaydedicisinin 20 ps'lik gecikmeye neden olacağını kabul edersek; sistemin 3 aşamalı pipeline'a sahip bir yapıya dönüştürülmesi durumunda elde edilebilecek

a) en yüksek

b) en düşük

throughput değerleri GIPS (Giga Instructions Per Second, Giga cinsinden Saniyede İşletilen Komut Sayısı) cinsinden ne olur?



Soru 3: (15 Puan) 9-bitlik float sayıların IEEE 754 standardına uygun şekilde gösterildiği bir sistemde anlamlı kısım (frac) için 4 bit ayrılmıştır. Aşağıdaki tabloda eksik bırakılan yerleri verilen açıklamalara uygun şekilde doldurunuz. ”—” içeren hücreleri doldurmanız istenmemektedir. Tablodaki sütunlara ait bilgiler şu şekildedir:

- **Tanım:** İlgili sayıyı ayırt edici şekilde tanımlayan ifade. Özel tanımlayıcı bir ifadesi olmayan sayılar için ”—” kullanılmıştır.
- **İkili:** İlgili sayının 9-bit ikili gösterimi.
- **M:** Anlamlı kısmın (frac) değeri. x ve y birer tam sayı; y , 2'nin bir kuvveti olmak üzere, bu kısım x veya $\frac{x}{y}$ şeklinde **belirtilmelidir**. Örneğin $0, \frac{3}{4}$.
- **E:** Üst (exp) kısmının tamsayı değeri.
- **Değer:** Sayının nümerik değeri.

Tanım	İkili	M	E	Değer
—	0 1100 0101			
En küçük denormalize sayı (negatif)				
En büyük normalize sayı (pozitif)				
Negatif bir				-1.0
Negatif sonsuz		—	—	$-\infty$

Soru 4: (15 puan) Aşağıda verilen Y86-64 assembly kod parçasındaki **prog** alt programı çağrılmadan hemen önce, ana hafıza içeriği şekilde verildiği gibiyse ve **%rdi** kaydedicisinde **0x8** değeri bulunuyorsa, **ret** komutu işletilmeden hemen önce **%rax**, **%rdx** ve **%rdi** kaydedicilerinin içerikleri ne olur? Aşağıda verilen kutucuklara **hexadecimal (onaltılık) şekilde yazınız**. Ana hafızadaki her bir kutu ilgili adreste kayıtlı bulunan **8 byte'lık** veriyi göstermektedir.

```

prog:
    irmovq $1, %r9
    irmovq $8, %r8
    irmovq $0, %rax
    mrmovq (%rdi), %rdx
    andq %rdx, %rdx
    je Son
dongu:
    addq %r9, %rax
    addq %r8, %rdi
    mrmovq (%rdi), %rdx
    andq %rdx, %rdx
    jne dongu
Son:
    ret
        
```

Ana Hafıza

Adres	Değer
6	0x00
3	0x08
5	0x10
8	0x18
3	0x20
6	0x28
0	0x30
1	0x38
8	0x40
7	0x48

%rax

%rdx

%rdi

Y86-64 komut seti hakkında gerekli bilgiler:

irmovq V, rB	--> rB'ye V değerini kopyala (Reg[rB]=V)
mrmovq D(rB), rA	--> rA'ya Mem[Reg[rB]+D] değerini kopyala (Reg[rA]=Mem[Reg[rB]+D])
andq rA, rB	--> Reg[rA]= Reg[rA] & Reg[rB] (Durum kodlarını (ZF, SF, OF) etkiler.)
addq rA, rB	--> Reg[rA]= Reg[rA] + Reg[rB] (Durum kodlarını (ZF, SF, OF) etkiler.)
je Hedef	--> Eğer ZF=1 ise Hedef'e dallan
jne Hedef	--> Eğer ZF=0 ise Hedef'e dallan
ret	--> Yığından geri dönüş adresini çekerek o adrese git

Soru 5: Bir bilgisayardaki önbellek (cache) ile ilgili olarak şu bilgiler verilmektedir

- Hafızadaki her bir byte farklı bir adrese sahiptir.
- Hafıza erişimleri 1-byte'lık kelimeler (word) halinde gerçekleşmektedir.
- Fiziksel adresler 14-bit genişliktedir.
- Önbellek 2-yönlü (yollu) küme ilişkilendirme (2-way set associative) yöntemine göre ilişkilendirilmiştir. Blok boyutu 4 byte'tır ve 32 küme bulunmaktadır.

Önbelleğin ilk 8 kümesinin içeriği aşağıdaki tabloda, tüm sayıların değerleri hexadecimal (onaltılık) şekilde ifade edilerek verilmiştir. Geri kalan 24 kümedeki verilerin hiçbirin geçerli durumda olmadığı bilinmektedir.

2-Yönlü (Yollu) Küme İlişkilendirmeli Önbellek												
İndeks	Etiket(Tag)	Geçerli(Valid)	Byte 0	Byte 1	Byte 2	Byte 3	Etiket	Geçerli	Byte 0	Byte 1	Byte 2	Byte 3
0	09	1	86	30	3F	10	00	0	99	04	03	48
1	45	1	60	4F	E0	23	39	1	00	BC	0B	37
2	4B	0	2F	81	FD	09	4B	0	8F	E2	05	BD
3	06	0	3D	94	9B	F7	5C	1	12	08	7B	AD
4	67	1	06	78	07	C5	05	1	40	67	C2	3B
5	5C	1	0B	DE	18	4B	2E	0	B0	39	D3	F7
6	11	1	A0	B7	26	2D	00	0	0C	71	40	10
7	39	0	B1	0A	32	0F	2E	1	12	C0	88	37

a) (12 Puan) Aşağıdaki kutular, her biri 1 biti içerecek şekilde fiziksel adres formatını göstermektedir. Bu kutuların her birinin aşağıda verilenlerden hangisine ait bir biti ifade ettiğini, kutu içerisine iki harflik kısaltmalarını yazarak belirtiniz. **(Bitler ayrı ayrı puanlandırılmayacaktır; üçünden herhangi bir bit grubuna ilişkin puan alabilmek için ilgili grubun tüm bitlerini eksiksiz/fazlasız doğru yerleştirmeniz gerekir.)**

CO Blok ofset değeri

CI Önbellek küme indeksi

CT Ön bellek etiketi

13 12 11 10 9 8 7 6 5 4 3 2 1 0

--	--	--	--	--	--	--	--	--	--	--	--	--	--

b) (3 Puan) 171D fiziksel adresinin bitlerini aşağıdaki kutulara (her kutuya 1 bit) yazınız.

13 12 11 10 9 8 7 6 5 4 3 2 1 0

--	--	--	--	--	--	--	--	--	--	--	--	--	--

c) (15 Puan) 171D fiziksel adresindeki veriye erişilmek istenirse, aşağıdaki tabloda listelenen parametrelerin değerlerini bulunuz. Kayıp (Cache Miss) durumunda "Dönülen Byte" kısmına "-" yazınız.

Parametre	Değer
Byte ofseti	0x
Önbellek küme indeksi	0x
Önbellek Etiketi	0x
İsabet durumu (Cache Hit)? (Evet/Hayır)	
Dönülen Byte	0x

Soru 6: (20 Puan)

Aşağıda verilen Y86-64 kod parçası, iletme (forwarding) ünitesi bulunan ve pipeline mimarisinin tam olarak gerçekleştirildiği Y86-64 simülöründeki işlemcide çalıştırılıyor. I11 komutu çekilene (fetch edilene) kadar çekilen (fetch edilen) komutları sırasıyla yazınız. İşlemcinin dallanma tahminini daima dallanma gerçekleşecek varsayımıyla yaptığını kabul ediniz. **Yazdığınız komutlar puanlandırılırken yazdığınız ilk hatalı komut ve sonrası değerlendirilmeyecektir.**

```
prog:
    irmovq $4, %rax    # I0
    irmovq $8, %rbx   # I1

dongu1:
    subq %rax, %rbx   # I2
    je    dongu2      # I3
    irmovq $10, %rdx  # I4
    mrmovq 0(%rdx), %rcx # I5
    addq %rcx, %rdi   # I6
    jmp   dongu1      # I7

dongu2:
    irmovq $4 %rcx    # I8
    irmovq $5 %rcx    # I9
    irmovq $6 %rcx    # I10
    ret               # I11
```

Y86-64 komut seti hakkında gerekli bilgiler:

irmovq V, rB	--> rB'ye V değerini kopyala (Reg[rB]=V)
mrmovq D(rB), rA	--> rA'ya Mem[Reg[rB]+D] değerini kopyala (Reg[rA]=Mem[Reg[rB]+D])
subq rA, rB	--> Reg[rB]= Reg[rB] - Reg[rA] (Durum kodlarını (ZF, SF, OF) etkiler.)
addq rA, rB	--> Reg[rB]= Reg[rA] + Reg[rB] (Durum kodlarını (ZF, SF, OF) etkiler.)
je Hedef	--> Eğer ZF=1 ise Hedef'e dallan
jmp Hedef	--> Hedef'e şartsız dallan
ret	--> Yığılından geri dönüş adresini çekerek o adrese git

CEVAP:

I0, I1, ...

BAŞARILAR
Yrd.Doç.Dr. Fatih GÖKÇE