**\*\*\*5-) 12 x 11 boyutlu bir array de [8, 10] elemaninin byte addresi nedir?**

**Array in storage order i row-major dir. (Note: Bu sorunun cevabini bulmak icin**

**uc kabule (bilgiye) daha ihtiyaciniz var. Istediginiz kabulleri yapabilirsiniz.)**

**Kabul1=dizinin ilk elemanı[1,1] dir**

**Kabul2=adreslemeye ilk elemandan baslasın**

**Kabul 3=her eleman 2 bittir**

**11\*10\*2=220 //7. satıra kadar herbiti hesaplar**

**+ //**

**10\*2=240 //8.satırda 10.sutuna kadar hesaplar toplam=240**

**5-)**

**a-) Aşağıdaki programın çalışması sonucu $11 ve $13**

**registerlerinde hangi değerler bulunur?**

**.data**

**aa: .word 5**

**bb: .word 0:3**

**cc: .word 0:5**

**dd: .word 16**

**ee: .word 27**

**.text**

**la $12,aa**

**lw $11,32($12)**

**lw $13,4($12)**

**$11=16**

**$13=0**

**b-)13 x 15 boyutlu bir array de [6, 8] elemanının addresi nedir?**

**Array in storage order i row-major dir. (Note: Bu sorunun**

**cevabını bulmak icin üç kabule (bilgiye) daha ihtiyacınız var.**

**Istediğiniz kabulleri yapabilirsiniz.)**

**Kabul1=dizinin ilk elemanı[1,1] dir**

**Kabul2=adreslemeye ilk elemandan baslasın**

**Kabul 3=her eleman 2 bittir**

**5\*15\*2=150 //5. satıra kadar herbiti hesaplar**

**+ //**

**8\*2=166 //6.satırda 8.sutuna kadar hesaplar toplam=166**

**1-) IEEE floating point arithmetic formatı düşünüldüğünde**

**, 32 bitlik(single precision) hatasız olarak temsil edilebilecek**

**en küçük ve en büyük tamsayıları decimal düzende belirtiniz?**

**En küçük sayı**

**1 11111111 111 1111 1111 1111 1111=**

**En büyük sayı**

**0 11111111 111 1111 1111 1111 1111**

**1-)**

**a-) Single-precision (tek duyarlı) IEEE gösteriminde,**

**floating point toplama işlemi sonucu overflow olup**

**olmadığı nasıl tespit edilebilir?**

**Son bite gelen elde ile son bitten çıkan elde xor a sokulur.**

**sonuç 1 ise overflow var sonuç 0 ise oewr flow yok..**

**b-) 7-bit mantissa kullanmak suretiyle aşağıdaki**

**toplama işlemlerini yapınız?**

**i-) 2005+1**

**11111010101=1.1111010101\*2^10**

**0 10001001 (1) 111 1010 1010 0000 0000 0000**

**1=2^0**

**0 01111111 (1) 000 0000 0000 0000 0000 0000**

**0 10001001 (1) 111 1010 1010 0000 0000 0000**

**0 10001001 (0) 000 0000 0010 0000 0000 0000**

**++++++++++++++++++++++++++++++++++**

**0 10001001 (1) 111 1010 1100 0000 0000 0000**

**ii-) 18.25+5200**

**4-) Aşağıdaki kodu SAL da yazin?**

**for i:=1 to 20 do**

**begin**

**a:=a+b;**

**if(a<5) then**

**c:=a;**

**end; .data**

**a .word**

**b .word**

**c .word**

**i .word 1**

**.text**

**For:**

**add a,a,b**

**add i,i,1**

**bgt i,20,endfor**

**bge a,5,for**

**move c,a**

**endfor:**

**1-) Aşagidaki kod parçacığı verilyor.**

**rol x,y,31**

**bgez y, else**

**or x,x,0x80000000**

**b next**

**else: and x,x,0x7FFFFFFF**

**next:**

**a-) Bu kod parçacığı ne yapar.**

**Arithmetic shift right 31 bit..**

**b-) Yukarıdaki işi tek başına yapabilen bir SAL**

**instruction ı varsa bunu yazin.**

**SRA**

**2-) Aşağidaki işlemleri ikinin tümleyenine göre yapın. Hangi sonuçların overflow olduğunu nedeniyle belirtin.**

**01110011 1101 1100 1101 0010**

**00111010 0101 1111 1100 0100**

+

+

+

**10101011 1 0011 1011 1 1001 0110**

**Var yok yok**

**Oveflowdan kurtulmak icin ne önerebilirsiniz?**

**Bit sayısını arttırmak….**

**3-) Aşağıdaki dizilerin 12. elemanının adresini hesaplayan SAL kodu yaziniz.**

**a)**

**chars: .byte 0:60**

**la $14 chars**

**move??(emin değilim bundan sülo soracak) $13, 11($14)**

**b-)**

**ints: .word 0:25**

**la $14 chars**

**move??(bundanda emin değilim) $13, 44($14)**

**4-)**

**.data  
ar: .word 0:50**

**.text**

**la $14, ar  
loop: li $8, 1**

**li $9, 0  
for: add $10, $14, $9   
 lw $11, ($10)   
 lw $12, 4($10)   
 sub $13, $11, $12  
 blez $13, noswap   
 li $8, 0  
 sw $11, 4($10)   
 sw $12, ($10)**

**noswap: add $9, $9, 4   
 sub $13, $9, 196**

**blz $13, for  
 beq $8, $0, loop   
done**

**Yukaridaki kod BubleSort işlemi yapmaktadır. Buna göre;**

**a-) Bu kodun kötü yanı nedir, sebebiyle belirtin?**

**Her seferinde dizinin son elemanına kadar control yaptığı için.**

**b-) Kodun yukarida belittiğiniz kötü yanını nasıl giderebilirsiniz? Yeni kodu yazınız?**

**.data  
ar: .word 0:50 // ar dizisini oluşturup adresini başlangıç ar a attı**

**.text**

**la $14, ar //14. registera ar ın başlangıç adresini**

**li $7, 200 // dizi uzunluğunu bit cinsinden sik..  
loop: li $8, 1 // 8. register a 1 ata**

**li $9, 0 // 9 register a 0 ata**

**sub $7 , $7 , 4 // her başa dönüşte dizinin sonuna bakma  
for: add $10, $14, $9 // 10 = 9 + 14   
 lw $11, ($10) // 10 u 11 e ata   
 lw $12, 4($10) // 10un 4 fazlasını 12 ye ata   
 sub $13, $11, $12 // 13 = 11 - 12   
 blez $13, noswap // 13 0 veya küçükse noswap a git**

**//değiştirme başla  
 li $8, 0 // 8 e 0 ata  
 sw $11, 4($10) // 11 = 11   
 sw $12, ($10) // 12 ye 10 u ata**

**// değiştirme bitir**

**noswap: add $9, $9, 4 // 9 a 4 ekledik  
 sub $13, $9, $7 // son elemandamısın**

**blz $13, for // son elemandamısın  
 beq $8, $0, loop   
done**

**1 5 3 8 4 7**

**Ekeleme:Biased neden kullanılır??**

**Sw ile Lw ile la arasındaki fark**