

Ad/Soyad:
Numara:

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
PROGRAMLAMA DİLLERİNİN PRENSİPLERİ II. ÖĞRETİM
2013 – 2014 Bahar Dönemi Final Sınavı

03.06.2014
Süre: 80 dakika

1. Aşağıdaki programın çıkışını dinamik kapsam bağlama ve statik kapsam bağlama kuralına göre bulunuz. (10 p)

```
procedure big
  var x: integer;
  procedure sub1;
    begin {sub1}
      print x;
    end; {sub1}
  procedure sub2;
    var x: integer;
    begin {sub2}
      x = 10;
      sub1();
    end; {sub2}
  begin {big}
    x = 3;
    ....
    sub2();
    ....
  end {big}
```

Cevap:

2. Aşağıdaki gramer için söz dizim grafiğini çizin. (10 p)

<if_stmt> → if <koşul> then <stmts {<else_if>} [else <stmts>] end if
<else_if> → elsif <koşul> then <stmts>

Ad/Soyad:
Numara:

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
PROGRAMLAMA DİLLERİNİN PRENSİPLERİ II. ÖĞRETİM
2013 – 2014 Bahar Dönemi Final Sınavı

03.06.2014
Süre: 80 dakika

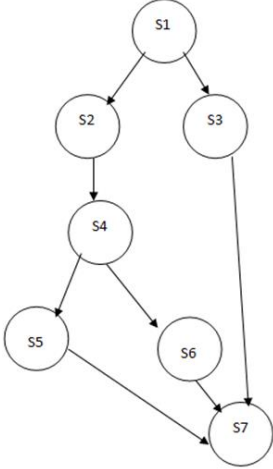
3. Aşağıdaki program parçası yapısalılığı içermeyen bir dile göre yeniden yazınız. (Goto deyimi, sadece if-then yapısı,) (5 p)

<pre>if ((c>7) OR (b==e)) then a=3 *(d+e) else a=c*(d+2)</pre>	
---	--

4. a) Aşağıda gösterilen ParBegin-ParEnd yapısı ile gösterilen A,B,C,D,E,F,G,H deyimleri için bir öncelik grafi çiniz. (7 p)

<p>A ParBegin C Begin B ParBegin D E F ParEnd G End ParEnd H</p>	
--	--

4. b. Aşağıda verilen öncelik grafini Fork/Join yapısı ile gerçekleştiriniz. (8 p)



5. Aşağıdaki programı inceleyiniz. İlgili parametre geçiş yöntemlerine göre sonuçlarını yazınız. (10 p)

Program	Değer ile aktarım	Değer-Sonuç ile aktarım	Referans ile aktarım
<pre>begin integer n; procedure p(k: integer); begin n := n+1; k := k+4; print(n); end; n := 0; p(n); print(n); end;</pre>			

Ad/Soyad:
Numara:

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
PROGRAMLAMA DİLLERİNİN PRENSİPLERİ II. ÖĞRETİM
2013 – 2014 Bahar Dönemi Final Sınavı

03.06.2014
Süre: 80 dakika

6. a) Aşağıda yazılmış Java kodunu C++'a çeviriniz. Tasarım işleyiş ve çıktı aynı olmalıdır. Değişkenlerin oluştuğu bellek bölgeleri aynı olmalıdır. (15 p)

Java

C++

<pre>public interface Ders { public String Adi(); public double Odev(); public double Vize(); public double Final(); }</pre>	
<pre>public class BilgisayarDers implements Ders { private String adi; private double odev; private double vize; private double fnl; public BilgisayarDers(String adi,double odev,double vize,double fnl){ this.adi = adi; this.odev = odev; this.vize = vize; this.fnl = fnl; } @Override public String Adi(){ return adi; } @Override public double Odev(){ return odev; } @Override public double Vize(){ return vize; } @Override public double Final(){ return fnl; } @Override public String toString() { String ekran = "Ders Adı:"+adi+"\nOdev: \t"+odev+"\nVize: \t"+vize+"\nFinal: \t"+fnl; return ekran; } }</pre>	
<pre>public static void main(String[] args) { // TODO code application logic here BilgisayarDers bd = new BilgisayarDers("PDP",40,20,40); System.out.println(bd); }</pre>	

6. b) Diamond problemi neden kaynaklanmaktadır. Java ve C++'ta bu problem ortaya çıkabilmekte midir? Nedeni ile birlikte yazınız. (5 p)

Ad/Soyad:
Numara:

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
PROGRAMLAMA DİLLERİNİN PRENSİPLERİ II. ÖĞRETİM
2013 – 2014 Bahar Dönemi Final Sınavı

03.06.2014
Süre: 80 dakika

7. Aşağıda Java’da yazılmış kodu C++ programlama dilinde tekrar yazınız. Hata tasarımı, işleyiş ve çıktı aynı olmalıdır. Değişkenlerin oluştuğu bellek bölgeleri aynı olmalıdır. (15 p)

<pre>public class TembelOgrenciHatasi extends NullPointerException { public TembelOgrenciHatasi(String mesaj){ super(mesaj); } public TembelOgrenciHatasi(){ super("Tembel Öğrenci"); } }</pre>	
<pre>public class Degerlendirme { private double kanaatNotu; private final int derseDevam; public Degerlendirme(int dersedevam){ kanaatNotu=0; derseDevam=dersedevam; } public void KanaatNotuVer(double not) throws TembelOgrenciHatasi{ if(derseDevam < 7) throw new TembelOgrenciHatasi(); kanaatNotu = not; } }</pre>	
<pre>public static void main(String[] args) { Degerlendirme d = new Degerlendirme(5); try{ d.KanaatNotuVer(70); } catch(TembelOgrenciHatasi ex){ System.out.println(ex.getMessage()); } }</pre>	

8. Aşağıda Java’da oluşturulan Fatih ve Mehmet nesneleri 1’den 10’a kadar olan kaynakları alıyorlar. Thread kullanıldığı için alımların sırası belli olmamaktadır. Kural ise 10’nuncu kaynağı ilk alan kazanacak ve ekrana kazanan olarak sadece onun adı yazacaktır. Kod bloğunda aşağıda boş bırakılan yerleri doldurunuz. Koda uygun yere Thread.sleep ekleyiniz. (15 p)

<pre>public class KaynakAl implements Runnable{ public String isim; public int kaynak; public KaynakAl(String isim){ this.isim = isim; kaynak=0; } @Override public void run() {</pre>	<pre>public static void main(String[] args) { ExecutorService havuz = KaynakAl fatih = new KaynakAl("Fatih"); KaynakAl mehmet = new KaynakAl("Mehmet"); KaynakAl kazanan = new KaynakAl(""); while(!havuz.isTerminated()){ if(fatih.kaynak == 10){ kazanan = fatih; break; } if(mehmet.kaynak == 10){ kazanan = mehmet; break; } } }</pre>
--	---

Önemli: Her sayfada mutlaka adınız yazmalıdır. Her soruyu altında veya yanında bırakılan boş yerlere cevaplayınız. Ek kâğıt verilmeyecektir.