



KARADENİZ TEKNİK ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
Bilgisayar Mühendisliği Bölümü
2012-2013 Eğitim-Öğretim Güz Yarıyılı
Windows Programlama Bütünleme Sınavı Soruları



- ✓ Soru kâğıdına **adınız, soyadınız** ve **numaranız** dışında başka hiçbir şey yazmayınız.
- ✓ Sınav süresi 120 dakikadır.

```
typedef CPoint CNokta;  
class CCizgi : public CObject  
{  
    DECLARE_SERIAL (CCizgi)  
public:  
    CCizgi (); // Varsayılan yapıcı  
    CCizgi (CNokta, CNokta); // Parametrelili yapıcı  
    void Ciz (CDC*); // Pencerenin (istemci) alanında bir önceki noktadan bir sonraki  
                    // noktaya çizgi çizme işlemi  
    virtual void Serialize (CArchive&); // Serileştirme işlemi  
    const CNokta* onceNoktaAl () const; // Bir önceki noktanın adresini geri döndürme  
    const CNokta* sonraNoktaAl () const; // Bir sonraki noktanın adresini geri döndürme  
    virtual ~CCizgi (); // Yıkıcı  
private:  
    CNokta m_nktOnce; // Bir önceki nokta  
    CNokta m_nktSonra; // Bir sonraki nokta  
};
```



Soru 1:

Yukarıdaki resimlerde gösterildiği gibi, bir resmin belli bir bölgesinin serbest seçimi yapılacaktır (Yöntemde, seçili bölgeyi kopyalama, taşıma veya yapıştırma gibi işlemler dikkate alınmayacaktır). Baş taraftaki resimde, fare aygıtının sol tuşuna basılı iken imleç hangi konuma taşınmışsa (sürüklenmişse), bir önceki konumundan o konuma kadar bir çizgi çizdirilmiş ve farenin sol tuşu bırakıldığında da çizim işlemi sonlandırılmıştır. Baş taraftakinden bir sonraki resimde ise, farenin sol tuşuyla resim üzerinde hangi konumlara tıklama yapılmışsa, herhangi bir dizide tutulan ve peş peşe olan bu ilgili konumlar bir doğruyla birleştirilerek bir çokgensel çizim yapılmıştır. Sondan bir önceki resimde ise, baş taraftaki resimdeki seçme işlemine benzer bir yaklaşım gerçekleştirilmiştir. Burada, dikkat edilmesi gereken husus ise, seçim, imleç konumuna eğer varsa en yakın imge kenarlarından otomatik olarak yapılmış ve böylece daha doğruluklu bir seçme işlemi gerçekleştirilmiştir (Yöntemlerde seçili bölgelerin kapalı olduğuna dikkat ediniz). İlgili çizim işlemlerde, eğer gerekli ise, yukarıda tanımlanmış yapılmış `CCizgi` sınıfına ait nesnelerin kullanıldığı ve `Belge/Görünüm` mimarisini destekleyen bir MFC tabanlı uygulamada seçilen noktaların bir `CObArray` tipindeki veya `CNokta` gibi herhangi uygun bir tipteki dizide tutulduğu varsayılın. Buna göre aşağıdaki soruları cevaplayınız.

- `void CGorunum::OnDraw (CDC*)` üye fonksiyonunda (uygulama penceresinin istemci alanında) seçili noktalar için bir çizim işlemi gerçekleştirilsin. Bu durumda, ilgili program kod parçalarını C/C++ programlama dilinde yazınız. (20 p)
- Yukarıda açıklandığı gibi, bir seçim işleminde, çokgensel bir bölge çizimi gerçekleştirilsin. Bu durumda, `void CGorunum::OnLButtonDown (UINT, CNokta)` üye fonksiyonunda gerekli olacak program kod parçalarını C/C++ programlama dilinde yazınız. (20 p)
- Yukarıda sondan bir önceki resimde gösterildiği gibi, ilgili bir resim bölgesinde akıllı bir seçim işlemi gerçekleştirilsin. Bu durumda nasıl bir algoritma geliştirilebilir? Kısaca açıklayınız. (15 p)
- Yukarıda en son resimde gösterildiği gibi, seçili bir bölgenin içersinin doldurulması işlemi gerçekleştirilsin. Uygulamada, herhangi bir üye fonksiyonun tanımlanması yapıldığı varsayılarak, bu üye fonksiyonun gövdesinde ilgili işlem için gerekli olacak kod parçalarını C/C++ programlama dilinde yazınız. (30 p)

Soru 2:

`Belge/Görünüm` mimarisini destekleyen MFC tabanlı bir veritabanı uygulaması oluşturulduğu varsayılın. Bu durumda, `Edit Control` gibi diyalog kontrolleriyle veritabanı alanları (tablo özellikleri) arasındaki veri alışverişinin (değiş tokuşunun [exchange]) nasıl yapıldığını şekil çizerek kısaca açıklayınız. (15 p)



KARADENİZ TEKNİK ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
Bilgisayar Mühendisliği Bölümü
2012-2013 Eğitim-Öğretim Güz Yarıyılı
Windows Programlama Bütünleme Sınavı Soruları



C.1) a)

```
int cizgiSayac = GetDocument()->cizgiSayiAl();
if (cizgiSayac)
{
    int cizKonum;
    CCizgi *cptCizgi;
    CDC *pDC= GetDC();
    for (cizKonum = 0; cizKonum < cizgiSayac; cizKonum++)
    {
        cptCizgi = GetDocument()->cizgiAl(cizKonum);
        cptCizgi->Ciz(pDC);
    }
}
```

b)

```
int m_nKonumIndeks;
CNokta m_pKonum[200];
//...
m_nKonumIndeks = 0; //ilgili yapıcı fonk. ilk değer atama yapılabilir.
//...
void CGorunum:: OnLButtonDown(UINT nBayraklar, CNokta nokta)
{
    CBelge *pBelge = (CBelge *) GetDocument();

    pBelge->m_pKonum[pBelge->m_nKonumIndeks] = nokta;
    pBelge->m_nKonumIndeks++;
}
```

c)

İmgede kenar algılaması yapılır ve farenin tıkladığı noktanın etrafında MxN boyutlarında bir pencerenin içerisindeki kenar algılanması yapılmış pikseller kontrol edilir, eğer ilgili pikseller kenarı temsil ediyorsa seçili nokta olarak işaretlenebilir.

d)

```
CNokta* cpt = new CNokta[cizgiSayac+1];
CCizgi* cptCizgi;
int cizgiKonum;
for (cizgiKonum = 0; cizgiKonum < cizgiSayac; cizgiKonum++) {
    cptCizgi = GetDocument()->cizgiAl(cizgiKonum);
    cpt[cizgiKonum].x = cptCizgi->onceNoktaAl()->x;
    cpt[cizgiKonum].y = cptCizgi->onceNoktaAl()->y;
}

cpt[cizgiKonum].x = cptCizgi->sonraNoktaAl()->x;
cpt[cizgiKonum].y = cptCizgi->sonraNoktaAl()->y;

CRgn bolge;
bolge.CreatePolygonRgn(cpt,cizgiSayac+1, ALTERNATE); //WINDING de olabilir.

CBrush firca( RGB(128,128,128) );
dc.FillRgn(&bolge, &firca);
delete[] cpt; cpt = NULL;
```

C.2) Aşağıdaki şekil verilebilir:

