PAİNT UYGULAMASI

Özet

### Paint bilgisayar üzerinde kullanıcıya basit grafik arayüzü sağlayarak üzerinde farklı grafik çizimlerinin gerçekleştirilmesini ve bu çizimlerin kullanıcının istediği boyut ve renklerle şekillendirilmesi, yapılan işleminjpeg,bmp gibi farklı resim formatlarında kaydedilmesini ve tekrar üzerinde düzenleme yapılabilmesini sağlayan bir çizim kitidir.

# Giriş

Günümüzde kullanıcıların farklı isteklerini karşılayabilmeleri için çeşitli programlar geliştirilmektedir. Paint uygulaması ile kullanıcının temel grafiksel ihtiyaçlarının karşılanması amaçlanmaktadır.

# Temel Bilgiler

Projenin tasarlama aşamasında Code Blocks idesi kullanılarak, Graphics.h kütüphanesi ile arayüz tasarımı ve bitmap\_image.hpp kütüphanesiyle ara yüz resimlerinin programa aktarılması sağlanmıştır.

# Geliştirilen Mimari

3.1 Kullanılan sınıfların ve metotlarının kısa açıklamaları

void arayuz(): Programın arayüz işlemlerinin yapıldığı metoddur.

void kaydet\_format\_bmp(): Kullanıcının yaptığı resmi bmp formatında kaydetmesini sağlayan metoddur.

void yukle(): Kullanıcının kaydettiği resmin tekrar program arayüzüne yüklenmesini sağlayan metoddur.

void dikdortgen(): Kullanıcının belirlediği büyüklük ve renkte dikdörtgen çizmesini sağlayan metoddur.

void kalem(): Kullanıcının belirlediği büyüklük ve renkte serbest çizim yapabilmesini sağlayan metoddur.

void cizgi(): Kullanıcının belirlediği büyüklük ve renkte düz bir çizgi çizmesini sağlayan metoddur.

void cember(): Kullanıcının belirlediği büyüklük ve renkte çember çizmesini sağlayan metoddur.

void ucgen(): Kullanıcının belirlediği büyüklük ve renkte üçgen çizmesini sağlayan metoddur.

void silgi(): Kullanıcının yaptığı çizimde silmek istediği yerlerin silinmesini sağlayan metoddur.

void arkaplandegistir(): Arka plan renginin değişmesini sağlayan metoddur.

3.2 Karşılaşılan problemler ve çözüm yaklaşımları

Karşılaşılan en büyük problem mousea tıklandığında tıklama durumlarının güncellenememesiydi. Bu problem menü alanında double click metodu kullanılarak çizim alanında ise single click metodu kullanılarak giderildi.   
  
2.problem ise örneğin bir dikdörgenin boyutlarının çizim aşamasında sabit boyut yerine mouseun hareketine göre büyütülüp küçültülmesiydi. Bu problem çizim işlemi while döngüsüne alınarak mouseun koordinatı her değiştiğinde bir önceki çizim işlemi arka plan renginde tekrar çizilip ekrandan silinmesi sağlandı ve yeni koordinata göre yeni bir çizim işlemi yapılarak giderildi.

3.3 Yazılım Geliştirme İçin Harcanan Süreler (kişi ve saat bazında)

Mehmet Samet Yıldız: 45 saat

Adem Derviş Yılmaz: 40 saat

3.4Dosyalama sisteminin, örnek veriler üzerinden anlatımı

Kaydetme işlemi için çizim alanının her pixeldeki renk değeri döngü ile tek tek alınarak bir txt dosyasına depolandı.

Yükleme işlemi için ise pixellerin renk değerlerinin kaydedildiği txt dosyasından renk değerleri okunarak ekrana pixel olarak tekrar yazıldı.

# Kullanıcı Kataloğu

Kullanıcı kataloğu için [tıklayınız.](#_Şekiller)(şekiller başlığı altındadır. Şekil 1)

# Sonuçlar

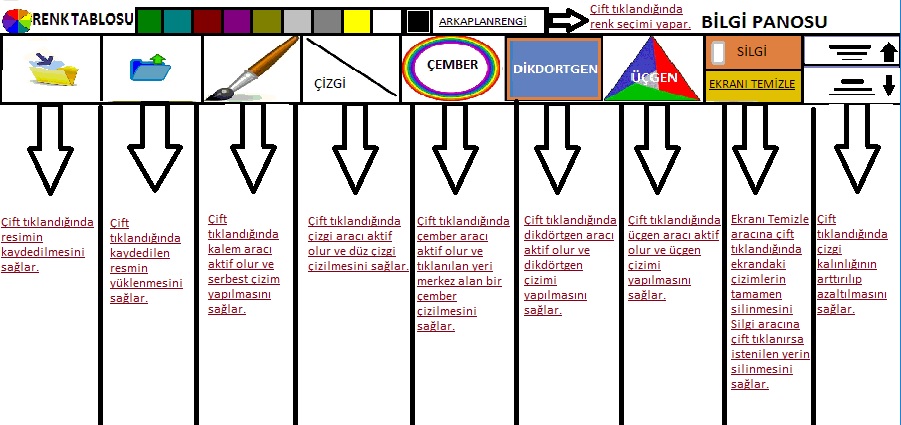
Uygulama içerisinde bulunan tüm fonksiyonlar istenildiği şekilde çalışmaktadır. Mouse hareketine göre otomatik boyutlandırma yapılmaktadır. Ayrıca kaydetme ve yükleme işlemleri de sağlıklı bir şekilde çalışmaktadır. Projemizdeki tek problem daha önceden çizim yapılan bir alanın üzerinden yeniden geçildiğinde bu alandaki bazı pixellerin arka plan rengine dönüşmesidir.

# Kaynakça

1. Yazar soyadı, Adının ilk harfi ve varsa aynı şekilde diğer yazarlar, “Çalışmanın adı”, Çalışmanın yayınlandığı dergi (veya internet sayfası gibi), yayın yılı.
2. Smith,J.O.andAbel,J.S.,``BarkandERBBilinearTrans­forms'',*IEEETrans.SpeechandAudioProc*.,7(6):697-708,1999.

# Şekiller, Denklemler ve Algoritmalar

## Şekiller



Şekil1:Uygulamanın kullanım kılavuzu

## Denklemler

Üçgenin koordinat denklemi: (1)

(x1,y1)

a b

(x1-(x2-x1),y2 )c ( x2,y2)

a=Line(x1-(x2-x1) ,y2,x1,y1)

b=Line(x1,y1, x2,y2)

c=Line(x1-(x2-x1),y2,x2,y2)

Dikdörtgenin koordinat denklemi: (2)

( x1,y1) a (x2,y1)

b c

(x1,y2) d (x2,y2)

a=Line(x1,y1,x2,y1)

b=Line(x1,y1,x1,y2)

c=Line(x2,y1,x2,y2)

d=Line(x1,y2,x2,y2)

Çemberin koordinat denklemi: (3)

(x1,y1)

r

(x2,y2)

r = y2-y1

çember=circle(x1,y1,r)

Çizginin koordinat denklemi: (4)

d=Line(x1,y1,x2,y2)

(x1,y1)(x2,y2)

A d B

Silginin koordinat denklemi: (5)

5

-5 0 5

- 5

Arkaplan rengi

Silgi

for(i=-5;i<5;i++)

{

for(j=-5;j<5;j++)

{

putpixel(mousex()+i,mousey()+j,arkaplanrengi);

}

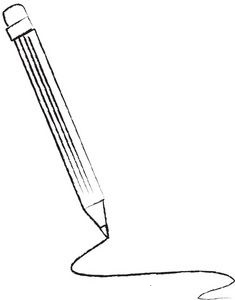
}

Kalem koordinat denklemi (6)

moveto(x1,y1);

lineto(x2,y2);

*"baslangıç değerlerine göre koordinat alınıyor ve verilen bitiş noktasına birleştirilerek çizim işlemi tamamlanıyor."*



## Algoritmalar

Program çalışma prensibi olarak ;

1) Main Fonksiyonun çalışması ve arayüzün ekrana getirilmesi

2) Arayüzden seçilen her işlem için fonksiyonların çağırılması

Olmak üzere 2 ana hat üzerine kurulmuştur.

Bu iki algoritma aşağıdaki akış diagramlarında belirtilmiştri. (Ek1,Ek2)

1)Main Fonksiyonun Çalışması ve Arayüzün ekrana getirilmesi

başla

evet

hayır

bitir

Fonksiyon sonuçlarını çizim ekranına bas

evet

hayır

Menu alanında sağ buton çift tıklandıysa

Mouse’un bulunduğu alandaki Fonksiyonu çağır

While(1)

Grafik Ekranını Tanımla

Karşılama Ekranı Fonk. Çağır()

Arayüz Fonk. Çağır()

Arkaplan Fonk. Çağır()

**Foksiyon Akış Diagramları**

başla

Bilgi Alanını Güncelle

Buton bilgisini güncelle

Çizim alanında Sol tuşa tıklandıysa

hayır

evet

hayır

evet

• Renk bilgisi Arkaplan rengine eşitle

* Bir önceki çizimi yeniden çizerek ekrandan silinmesini sağlar
* Mouse bitiş koordinat bilgilerini yeniden al
* Renk bilgisini güncelle
* Yeniden çizim yap

Mouse hareket halinde mi?

hayır

bitir

• Mouse bitiş koordinatını bilgilerini değişkenlere ata

* Renk bilgisini al
* **Seçilen çizim fonksiyonu için gerekli olan Başlangıç ve Bitiş değerlerine göre çizim işlemini gerçekleştir.**

evet

WHİLE(Mouse Çizim alanında ise ve halen sol buton basılı ise

* Mouse başlangıç koordinatını bilgilerini değişkenlere ata
* Buton bilgilerini güncelle