5. Resim Ekleme

Bu başlıkta öğreneceğiniz konular

* Resim Ekleme
* Resimleri Mask(Kesme-Kırpma) Etme
* Resimleri Döndürme

## Dosyadan Resim Ekleme

WinBGI penceremizde resim dosyalarını göstermemize izin verir. BMP, JPG, GIF formatında resim ekleyebilirsiniz. Peki, nasıl ekleriz?

En kolay yolu **readimagefile()** fonksiyonunu kullanarak resim eklemedir.

# void readimagefile( const char\* filename=NULL, int left=0, int top=0, int right=INT\_MAX, int bottom=INT\_MAX  );

|  |  |
| --- | --- |
| **D:\A-Computer Engineer\WEB YARDIMCIMIZ\YAZILAR WORD\C++ da Graphic.h Kütüphanesi Ve Oyun Yapımı\5.Resim Ekleme\5_dosyalar\image001.png** | filename:eklemek istediğimiz resmin bilgisayardaki konumudur (Örnek “D:\\resim1.bmp”). Eğer koymak istediğiniz resim uygulamanın alt klasöründe ise **“resimadı.formatı”** diye kullanmanız yeterlidir. Uygulama konumunun alt klasöründe ise **“klasöradı\\resimadı.formatı”** diye kullanmalısınız.  left:resmimizin sol tarafının yapıştırılacağı konumdur. Sol taraftaki X=100 dür.  top:resmimizin üst tarafının yapıştırılacağı konumdur. Üst  taraftaki Y=50 dir.  right:resmimizin sağ tarafının yapıştırılacağı konumu belirler. Sağ taraftaki X=217 dir.  bottom:remimizin alt tarafının yapıştırılacağı konumu belirler. Alt taraftaki Y=177 dir.    genişlik = right - left **,** yükseklik = bottom – top **dur.**    **readimagefile("resimler\\gunes.bmp",100,50,227,177);**    Yandaki örneği indirmek için **[D:\indir.bmp](http://bc.vc/DZjaGt)** |
| **D:\A-Computer Engineer\WEB YARDIMCIMIZ\YAZILAR WORD\C++ da Graphic.h Kütüphanesi Ve Oyun Yapımı\5.Resim Ekleme\5_dosyalar\image003.png** | Yandaki örneğimizde de resmi farklı konuma farklı boyutlarda yapıştırdık. İndirmek için **[D:\indir.bmp](http://bc.vc/DWASaw)**    **readimagefile("resimler\\gunes.bmp",120,20,200,80);** |

## Bellekten Resim Yapıştırma

readimagefile() fonksiyonunu animasyon hazırlarken kullanamayız. Çünkü bu fonksiyon her seferinde hard-disk ten verileri alır. Fakat verileri Hard-disk’ten alma hızı yavaştır. Bu yüzden RAM icat edildi ya. Hard-disk’ teki verileri RAM’ e atıp RAM den de hızlı bir şekilde kullanmalıyız. Bu sayede oyunumuz yada programımız hızlı çalışır. Peki  bunu nasıl yapıcağız?

****

Ekrana resim çizdireceğiz. Bu normal olarak çizgilerle dairelerle de olabilir. Yada üst başlıktaki gibi dosyadan da alabiliriz.



Bu kısımda ekranın belirli bölgesindeki resmi RAM’ e alıcağız.



#### void \*resim;

Burada biz bir pointer yani adres oluşturduk. Adres bellekteki konum demektir. Sınırı belleğinizin sınırı kadardır. Fakat şuanlık bir sınır yada gerekli yeri ayırmadık.



#### resim=malloc(imagesize(x1,y1,x2,y2));

Burada da biz (x1,y1) konumlarından (x2,y2) konumuna kadar olan kısımda olabilecek resmin boyutunu imagesize ile ölçüyoruz. Oluşturduğumuz adres resim içinde ölçtüğümüz resim boyutunu adreste ayırtıyoruz. Şuan resmi RAM’ e atmak için gerekli her şey tamam.



#### getimage(x1,y1,x2,y2, resim);

Şimdide resim adresine(pointer) resmimizin verilerini gönderiyoruz.



#### putimage(x,y,resim,yapıştırma\_metodu);

RAM’ deki resmimizi penceredeki x,y konumuna yapıştırıyoruz.

|  |  |
| --- | --- |
| YAPIŞTIRMA\_METODULARI |  |
| COPY\_PUT | Kopyasını yapıştırma |
| OR\_PUT | Resimlerin veya’ sını alır. Bir nevi renkleri toplar |
| AND\_PUT | Resimlerin ve’ sini alır. Resimdeki renklerin farkını alır. |
| XOR\_PUT | Resimlerin xor’ unu alır. |

### Örnek 3 [D:\indir.bmp](http://bc.vc/lxvmiw)

#include <graphics.h>

int main()

{

initwindow(400,100, "5. Resim Ekleme");

readimagefile("resimler\\gunes.bmp",0,0,100,100);

void \*resim;

resim=malloc(imagesize(0,0,100,100));

getimage(0,0,100,100, resim);

int x=-100;

while(1){

cleardevice();

putimage(x+=10,0,resim,COPY\_PUT);

swapbuffers();

if (x>500) x=-100;

delay(40);

}

closegraph();

return 0;

}

### Örnek 4: Saat Uygulaması [D:\indir.bmp](http://bc.vc/h9QUuv)

## Resimleri Mask(Görünmeyen Yer) Etme

Resimleri mask etmek; resmin tamamını değil de bazı bölgelerinin görüntülenmesini istediğimiz zaman kullanırız.

|  |  |
| --- | --- |
| Masksız saatimiz şu şekilde olurdu | Masklı saatimiz ise şu şekilde |
| C:\Users\Nöbetçi\Desktop\masksız.png | C:\Users\Nöbetçi\Desktop\masklı.png |

Mask nasıl yapılır?

1. Arka-Plan resmini ekrana yapıştırıyoruz.



1. Sonra resimlerin göstermek istediğimiz yerlerini OR\_PUT yöntemiyle siyah-beyaz olarak yapıştırıyoruz.

**Görünmesini** istediğimiz yerleri **beyaz**, **görünmemesini** istemediğimiz yerleri ise **siyah** renkli yapıyoruz.

 D:\A-Computer Engineer\121213040_Hasan_DELIBAS\dijim.bmp

1. Mask görüntüsün üzerine AND\_PUT yöntemi ile renkleri de aktarıyoruz.

Şimdide **görünmemesini** istediğimiz alanları beyaz renkle boyuyoruz.

 D:\A-Computer Engineer\121213040_Hasan_DELIBAS\diji.bmp

1. Sonuç olarak bu yöntemle eklemek istediğimiz yeri bu şekilde mask ediyoruz.

### Örnek 5: Mask Edilmiş Saat Uygulaması [D:\A-Computer Engineer\WEB YARDIMCIMIZ\YAZILAR WORD\C++ da Graphic.h Kütüphanesi Ve Oyun Yapımı\5.Resim Ekleme\5_dosyalar\image002.png](http://bc.vc/KMTnkR)

#include <graphics.h>

#include <stdio.h>

#include <math.h>

int main(int argc, char \*argv[])

{

void \*rsayi[10];

void \*rsayim[10];

void \*rarka;

initwindow(372,292,"5. Resim Ekleme : Saat Örneği",0,0);

readimagefile("diji.bmp" ,0,0,300,55);

for (int r=0;r<10;r++){

rsayi[r]= malloc(imagesize(r\*30 ,0,r\*30+30,55));

getimage(r\*30 ,0,r\*30+30,55, rsayi[r]);

}

readimagefile("dijim.bmp" ,0,0,300,55);

for (int r=0;r<10;r++){

rsayim[r]= malloc(imagesize(r\*30 ,0,r\*30+30,55));

getimage(r\*30 ,0,r\*30+30,55, rsayim[r]);

}

readimagefile("resim.jpg" ,0,0,372,292);

rarka= malloc(imagesize(0,0,800,600));

getimage(0,0,800,600, rarka);

int sure[3]={0,0,0};

int konumsaatx=300;

int konumsaaty=150;

while(!kbhit()){

putimage(0,0,rarka,COPY\_PUT); // Arka Plan Görüntüsü

for (int i =0; i<3 ;i++){

putimage(konumsaatx-i\*80-30,konumsaaty,rsayim[(int)sure[i]%10],OR\_PUT); // Mask Edilen Görüntü

putimage(konumsaatx-i\*80-30,konumsaaty,rsayi[(int)sure[i]%10],AND\_PUT); // Mask Üzerine Resmin Kendisi

putimage(konumsaatx-i\*80-60,konumsaaty,rsayim[(int)sure[i]/10],OR\_PUT);

putimage(konumsaatx-i\*80-60,konumsaaty,rsayi[(int)sure[i]/10],AND\_PUT);

}

// Süre Fonksiyonları

sure[0]++;

if (sure[0]==100){ sure[0]=0;sure[1]++;}

if (sure[1]==60){ sure[2]++;sure[1]=0;}

if (sure[2]==60){sure[2]=0;}

// Süre Fonksiyonları

swapbuffers();

delay(7);

}

return EXIT\_SUCCESS;

closegraph();

}

## Resimleri Döndürme

WinBGIm’ de resimleri özel olarak döndürme kodu yoktur. Resimleri kodlarla döndürmek istediğimiz zaman ise programımız yavaş çalışacaktır. Maalesef WinBGIm’ in basit bir motor kullandığı için biz resim döndürme işlemini üst konularda sayıları dizi içerisine attığımız gibi, resmin de (0,30,60,90 …) belirli açılardaki resimlerini dizimize atıp, cismin açısına göre dizideki resimleri kullanacağız.

Sizler için sıralı resim döndürme programı yaptım. Sıralı resim döndürme programı için. [D:\A-Computer Engineer\WEB YARDIMCIMIZ\YAZILAR WORD\C++ da Graphic.h Kütüphanesi Ve Oyun Yapımı\5.Resim Ekleme\5_dosyalar\image002.png](http://bc.vc/aUdOLS)



### Örnek Resmimiz: Sıralı Resim Döndürme Programı Çıktısı. Programı İndirmek İçin [D:\A-Computer Engineer\WEB YARDIMCIMIZ\YAZILAR WORD\C++ da Graphic.h Kütüphanesi Ve Oyun Yapımı\5.Resim Ekleme\5_dosyalar\image002.png](http://bc.vc/aUdOLS)



0 30 60 90 120 150 180 210 240 270 300 330

1. Resimleri ekrana yapıştır.
2. \*resim[12] dizisi oluştur.
3. Ekrandaki resimleri sırasıyla \*resim[i] dizisine at.
4. Dönen resimleri ekranda çizmek için putimage(konumx,konumy,resim[açı/12],yapıştıma metodu) kullan.

### Örnek 6: Hem Döndürülmüş hem de mask edilmiş uzay gemisi [D:\A-Computer Engineer\WEB YARDIMCIMIZ\YAZILAR WORD\C++ da Graphic.h Kütüphanesi Ve Oyun Yapımı\5.Resim Ekleme\5_dosyalar\image002.png](http://bc.vc/zLVMKS)



**Hayal gücünün sınırlarını aşma vakti geldi.**