

**SAKARYA ÜNİVERSİTESİ**  
**BİLGİSAYAR VE BİLİŞİM FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**VERİ YAPILARI DERSİ ÖDEV 1**

Ödev hiyerarşisini Sembol dosyaları en başta olmak üzere ardından Kart dosyaları ve en son da Menü dosyaları olmak üzere kurdum.

### **Sembol.hpp**

Bir "Sembol" sınıfı oluşturup, erişilmesi istemediğim değişkenler private adı altında tanımladım. Bu private değişkenler üzerinde direkt değişiklik yapılmaması adına ya da direkt çağırılmamaları dolayısıyla public adı altında private değişkenlere ait setter ve getter fonksiyonların referanslarını oluşturdum.

Sembol başlık dosyasında bulunan "windows.h" kütüphanesi kart renklerini ayarlayabilmek için yazım. Komut satırında renk kullanmak için işletim sistemi tabanlı bir çözüme başvurmam gerekti. Çünkü renk göstermek için genel bir çözüm yolu araştırmama rağmen bulamadım. Open source bir kütüphane olan ve konsol üzerinde renklendirme işlemleri yapmamı sağlayan "rutil" kütüphanesinden işime yarayacak fonksiyonların çalışma şekillerini anlayıp kendi projem için gerekli şekilde kullandım.

Aşağıdaki komutları da Windows renk kütüphanesi için oluşturdum.

```
string SembolBelirle(int&,int&);  
int& SembolRenkBelirle(int&);  
int SembolMevcutRenkTut();  
void SembolRenkResetle();
```

### **Sembol.cpp**

Setter ve getter fonksiyonları teker teker tanımlanarak private değişkenlere erişim sağladım.

SetSembolRenk() adlı fonksiyonun parametresinin integer dönüş tipinde olmasının sebebi, Windows'un renk skalasındaki her rengin bir sayısal karşılığı olmasından ve öyle çağırılmalarından kaynaklandı.

```
void Sembol::SetSembolRenk(int rasgeleSayi)  
{  
    this->sembolRenk=rasgeleSayi;  
}
```

### **Kart.cpp**

Kart sınıfından nesne oluşturulduğu anda sembol tipinin ve renginin rastgele belirlenmesi için gereken rastgele sayılar da nesne ile beraber kurucu fonksiyonda oluşturulmaktadır. Böylece her nesne için rastgele bir sembol ve rastgele bir renk ataması yapılabilinmektedir.

```
this->randTip=semboller->SembolTipiDondur()=rand()%4;  
this->randRenk=semboller->SembolRenkDondur()=rand()%16;
```

Simgeleri renklendiren fonksiyonun kullanımından sonra SembolRenkResetle() fonksiyonunun çağırılmasına ihtiyaç vardır. Aksi halde oluşturulan simgeden sonraki simgeler aynı renkte devam etmektedir.

### **Menuler.cpp**

Seçtiğim 4 konsol sembolünün konsol üzerinde görüntülenebilmesi için komut satırının charset ini 'utf-8' e çevirdim. Menüler.cpp dosyası içerisindeki system("chcp 65001") kodu, program çalıştırıldığında ilk bu fonksiyon çağırıldığı için tek seferlik bu işlemi yapmak amacıyla kullanıldı.

Menüler sınıfı, "int main()" in içerisinde yaptırabileceğimiz işlemleri daha derli toplu bir hale getirmek programımızı daha güvenli bir şekilde çalıştırabilmek için oluşturuldu.

Bu sınıf sayesinde kartlar ve kartlar üzerinden semboller için gerekli manipülasyonları yapabildik. Seçim yapmamızı sağlayan seçim menüleri de bu sınıf üzerinden yönetildi.

"iomani.h" kütüphanesi yardımıyla semboller ile kart numaralarını hizalamak amacıyla "setw" yi kullandım.

### **Sonuç**

Ödev boyunca devam eden hiyerarşi mantığı, ihtiyacımız olan kütüphaneleri kullanma gerekliliği bu ödevin bana kazandırdıkları oldu.

Ödevde benden çıktı olarak istenilenler gerçekleşmiştir.