|  |  |  |
| --- | --- | --- |
|  |  |  |

**T.C.**

**SİVAS CUMHURİYET ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

|  |  |  |
| --- | --- | --- |
|  | **BİL3011 BİL. G. GRAFİK PROGRAMLAMA**  **Çizgi Çekme Algoritmaları**  **Kesikli Çizgi Çizme**  **Türk Bayrağı**  **ÖDEV-2**  **24/10/2023**    **Çağatay ÜRESİN** |  |

**Ekim-2023**

**SİVAS**

**1. GİRİŞ**

**Bu rapor, Grafik Programlama dersinde verilen 2 numaralı ödevin nasıl yapıldığını içermektedir. Ödev açıklaması şu şekildedir:**

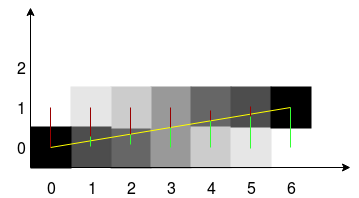
**Ödev 1- İki nokta arasında çizgi çizme algoritmalarından birisi seçilip incelenecek ve açıklanacaktır.**

**Ödev 2- Başlangıç ve bitiş noktaları kullanıcıdan alınarak iki nokta arasında kesikli, düz bir çizgi çizilecektir.**

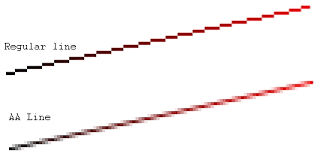
**Ödev 3- Standart bayrak ölçüleri orantılanarak kırmızı zemin üzerine Türk bayrağı çizilecektir.**

**2. YÖNTEM**

1- Xiaolin Wu’nun Çizgi Çekme Algoritması ilk ödev için tercih edilmiştir. Bu algoritma Wu’nun algoritmasının geliştirilmiş versiyonudur. Algoritmanın asıl başarısı çizgilerin yumuşatılmasındadır. Normalde örneğin siyah bir çizgi çekilmesi sırasında eğilen çizgi yakınlaştırıldığında merdiven görüntüsüne sahip olacaktır. Bu algoritma ise renklerin alfa kanallarını kullanarak eğilen kısma eğilme oranında bir hesap yaparak yumuşak pikseller ekler. Böylelikle çizgi arka planla daha rahat karışır.

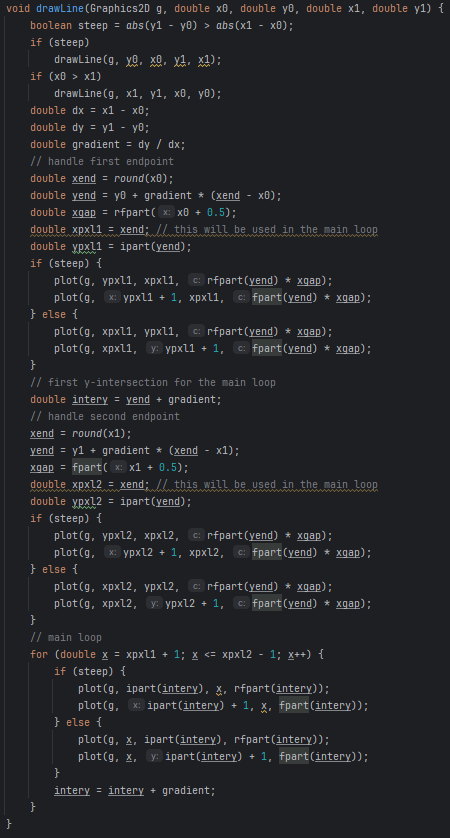


Antialiasing denilen bu işlem düşük çözünürlükteki görsellerde daha başarılı bir yumuşatma sağlar.



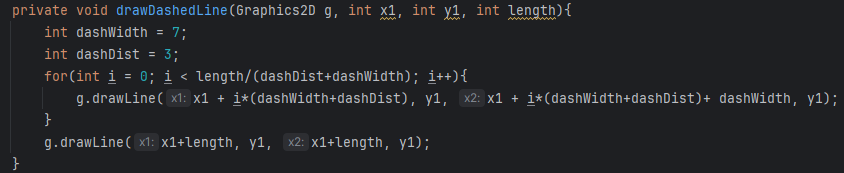
Şekilde görüldüğü üzere antialiasing çizgilerin daha pürüzsüz görünmesine yardımcı olur.

Xiaolin Wu Çizgi Çekme Algoritması için Java kodu aşağıdaki gibidir.



2- Başlangıç ve bitiş noktaları kullanıcıdan alınan kesikli düz çizgi çekme için öncelikle çizginin anatomisi belirlenmiştir. Bir çizgi ve çizginin boşluk kısmı toplamda 10px olduğunda çizili kısım 7px çizgisiz kısım 3px uzunluğuna tekabül edecektir.

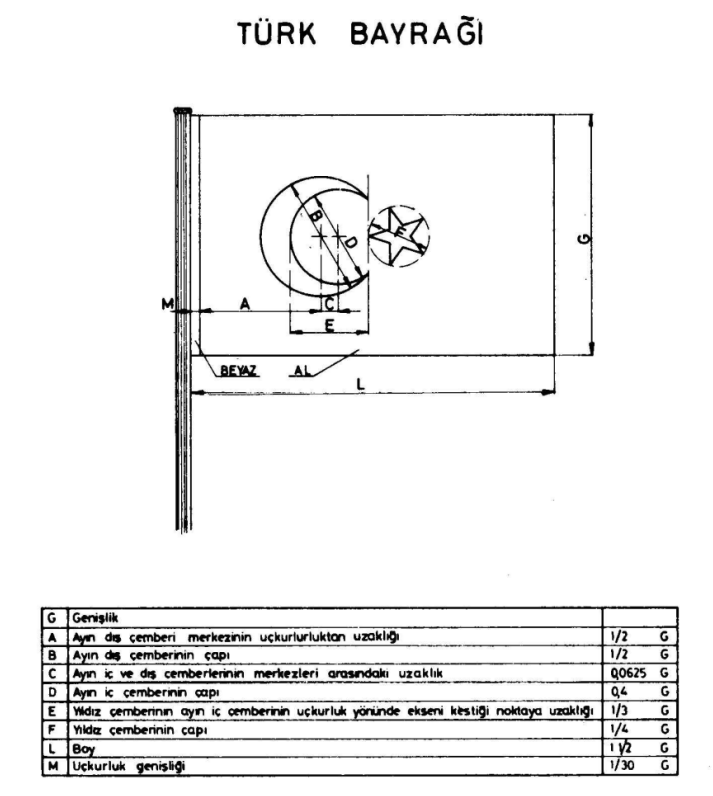
Yazılan metot çizginin başlangıç koordinatlarını ve uzunluğunu alıp soldan sağa kesikli çizgi çekecektir.



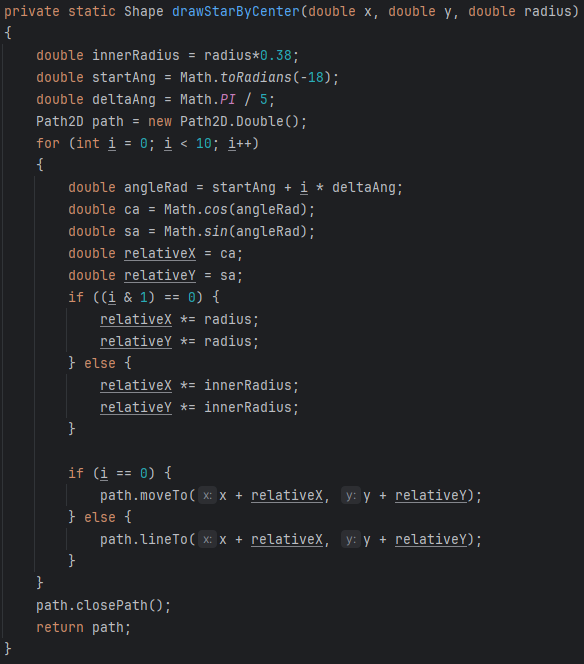
Metot başlangıçta belirttiğimiz gibi kendini 10px için hazırlamıştır. Böylece kaç defa çizgi çekmesi gerekiyorsa o kadar kere çalışacaktır. Bu metotun dezavantajı. Mod10'u 3’ten küçük olan uzunluklarda bittiği noktanın gerisinde çizgi çizilmemiş olacağından sonuçta kısa çizgiler çekmiş gibi görünecek olmasıdır. Bu durumdan kaçınmak için metot en sonda bir tane de bitiş noktasına pir piksel daha koyup kendini tamamlamaktadır.

Daha başarılı sonuçlar verebilecek metotlar geliştirilebilirdi, ancak ödev süresini iyi optimize edemediğim için ilkel bir versiyon ile bitiriyorum

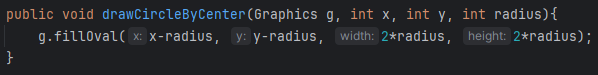
3- Türk Bayrağı Çizme ödevi için öncelikle 2893 Sayılı Türk Bayrağı Kanunu incelenmiştir. Türk Bayrağı’nın oranları kanunda aşağıdaki gibi belirtilmiştir.



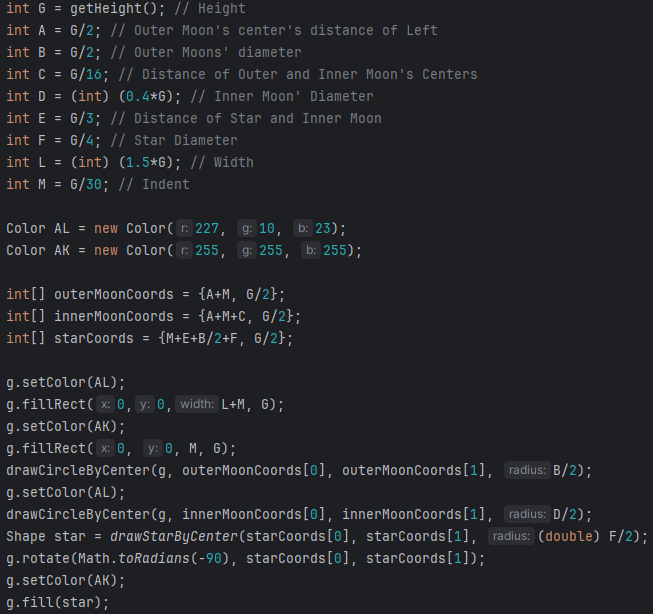
Yukarıda da görüldüğü üzere özellikle yıldız kısmı çizilirken Java Swing API’nin Path, LineTo, MoveTo komutları ve Shape nesnesi kullanılacaktır. Shape nesnesi oluşturulduktan sonra fill metodu ile rengi değiştirilebilir rotate ile döndürülebilir olacaktır. Yıldız için gerekli matematiksel formüllerden yararlanılarak. Merkez noktası ve çapı verilen yıldızı Shape nesnesi olarak döndürecek metot aşağıdaki gibidir.



Hilal kısmı dış ve iç daireden oluştuğundan merkez noktası ve çapı verildiğinde bize daire çizecek bir metoda ihtiyaç olacaktır. O metot aşağıdaki gibi yazılmıştır.

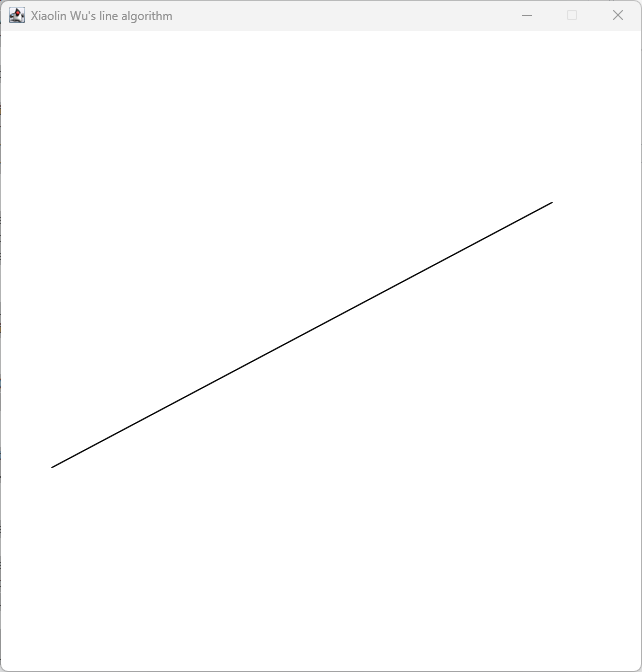


Çizim için çalışan ana kısımda değişkenler tıpkı kanundaki gibi belirtilmiş olup gerekli hesaplamalar yapılmış ve böylelikle çok daha kısa ve anlamlı bir kodla bayrak çizilmiştir.



**3. SONUÇ**

Xiaolin Wu’s Line Algorithm için verilen java kod parçacığının çıktısı:



Çizgili çizgi çekme algoritmamın çıktısı aşağıdaki gibidir.



Türk Bayrağı Çizme Algoritması’nın çıktısı da aşağıdaki gibidir.



**KAYNAKLAR**

[1] https://www.mevzuat.gov.tr/mevzuatmetin/1.5.2893.pdf

[2] Wu, Xiaolin (July 1991). "An efficient antialiasing technique". Computer Graphics. **25** 143 152. doi:10.1145/127719.122734. ISBN 0-89791-436-8.

[3] https://rosettacode.org/wiki/Xiaolin\_Wu%27s\_line\_algorithm