

Bildiri Başlığı (Türkçe) (*paper title*)

Title of Paper (In English)

Ahmet Kayra Gügercin
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli/Türkiye
220201012@kocaeli.edu.tr

Cüneyt Oğuz Başkurt
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli/Türkiye
220201051@kocaeli.edu.tr

Abstract—Bu doküman kocaeli üniversitesi bilgisayar mühendisliği bölümü programlama laboratuvarı dersi 1.proje odevi için hazırlanmıştır.

I. ÖZET

Bu projede sondaj faaliyeti yapan bir şirketin maksimum kâr elde edebilmek amacıyla arama bölgesini en doğru sayı ve optimal boyutlarla bölünmüş alanlara (parsellere) ayırmayı hedeflemektedir. Bu amaç için belirli kurallara göre kar-maliyet hesaplamalarıyla birlikte en uygun alan bölümlleme işlemini yapan ve bölümlenmiş arama alanlarının görsel haritasını oluşturan bir yazılım geliştirilmesi beklenmektedir.

II. GİRİŞ

Projenin amaçları şunlardır: CURL ile verilen web sitesi URL'sinden şekilleri ifade eden koordinatları çekip bu koordinatları konsola yazdırmak. Daha sonra SDL(Simple DirectMedia Layer) kullanarak bir arayüz oluşturmak ve bu arayüze çektiğimiz koordinatların şekillerini çizdirmek. Çizdirdiğimiz bu şekilleri birim karelere ayırdıktan sonra alanın hesaplanması. Bu şekilleri optimal boyutlarda karelere ayırarak kullanıcıdan aldığımız birim sondaj maliyeti ve birim platform maliyeti değerleri sayesinde toplam platform maliyeti ile toplam sondaj maliyetinin hesaplanması. Şeklin veya şekillerin toplam alanının on ile çarpılması sonucu elde edilen rezerv değerinin bulunması. Toplam platform maliyeti ve toplam sondaj maliyetinin toplanmasıyla toplam maliyetin hesaplanması. Şeklin veya şekillerin toplam alanının on ile çarpılması sonucu toplam rezerv değerinin bulunması. Toplam rezerv değeri ile toplam maliyetin çıkarılması sonucu elde edilen toplam kârın hesaplanması. Bu verilerin ve şekillerin ekranda gösterilmesi ve daha iyi kullanıcı deneyimi için bir arayüzün tasarlanması.SDL_TTF ile oluşturduğumuz bu arayüze yazı yazdırmamız.

III. YÖNTEM

Proje başlangıcında ilk olarak internet üzerindeki bir web sitesi URL'sinden veri çekmekte kullanılan CURL kütüphanesini indirdik. Projeyi geliştirdiğimiz CodeBlocks idesine indirdiğimiz bu CURL kütüphanesini entegre etmek için linker ve kütüphane bağlantılarını gerçekleştirdik.Projede kullanabilmek için bu kütüphaneyi kodumuzun içine include ettik.

CodeBlock idesinden bir arayüz oluşturmak için kullanılan SDL kütüphanesini indirdik. Projeyi geliştirdiğimiz CodeBlocks idesine indirdiğimiz bu SDL kütüphanesini entegre etmek için linker ve kütüphane bağlantılarını gerçekleştirdik.Projede kullanabilmek için bu kütüphaneyi kodumuzun içine include ettik.

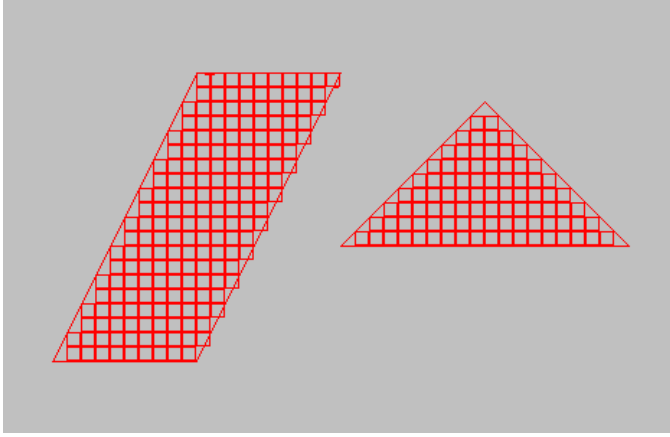
İlk olarak CURL kütüphanesini kullanarak bize verilen web site URL'sinden şekilleri ifade eden koordinatları aldık. Bu koordinatlar belli bir formatta gelir. Bu formatın özellikleri şunlardır: Her satırın başında satır sayısı ve her satırın sonunda satırın bittiğini ifade eden 'F' karakteri bulunur. Bu veriden çektiğimiz koordinatları her bir satırı ayrı bir diziye atmak için bu belirteçleri referans olarak veriyi parser (ayırıştırma) ettik. Daha sonra ayırdığımız bu verideki her satırı farklı bir diziye atadık.

Kullanıcının hangi satırdaki şekilleri çizdirmek istediğini öğrenmek için arayüzde ilgili satırlar için bir buton bölümü oluşturduk.

1. Satır	2. Satır	3. Satır	4. Satır	5. Satır
6. Satır	7. Satır	8. Satır	9. Satır	10. Satır
11. Satır	12. Satır	13. Satır	14. Satır	15. Satır
16. Satır	17. Satır	18. Satır	19. Satır	20. Satır

Daha sonra kullanıcının seçtiği satırdaki şekli çizdirmek için bir çizim algoritması oluşturduk. Bu butonlara basıldığında hangi satıra basılıysa o satırdaki şekilleri ekrana çizdirdik.

Şekilleri ayırt etmek için bize verilen formatta şeklin başlangıç ve bitiş koordinatları aynıdır. Bunu ayırt etmek için satır içinde tekrar eden koordinatları kontrol ettik. Bunun sonucunda şekilleri sırayla ekrana çizdirdik. Daha sonra çizdirdiğimiz bu şekillerin alanlarını hesap edebilmek için ray crossing birim karelere ayırdık.



Birim kareler yardımı ile çizdirdiğimiz bu şekillerin alanlarını hesapladık. Kullanıcıdan aldığımız birim sondaj maliyeti ve birim platform maliyetine göre bazı hesaplamalar yaptık. Bu hesaplamalar şunlardır: Rezerv değeri,platform sayısı,sondaj maliyeti,platform maliyeti,toplam maliyet,toplam kâr. Bu hesaplamaları 'Hesapla' adını verdiğimiz bir buton ile gerçekleştirdik ve bu değerleri arayüzde kullanıcıya göstermek için bir bölüm ayırdık. Hesapla butonu ayrıca şekli optimal boyutta karelere ayırmaktadır.

Sondaj Maliyeti : 1	Rezerv Değeri: 0
Platform Maliyeti : 1	Platform Sayısı: 0
	Sondaj Maliyeti: 0
	Platform Maliyeti: 0
Hesapla	Toplam Maliyet: 0
	Toplam Kar: 0

Oluşturduğumuz bu butonlar ve kutular yardımı ile kullanıcıya kullanışlı bir arayüz tasarladık.

1. Sıra	2. Sıra	3. Sıra	4. Sıra	5. Sıra	Sondaj Maliyeti : 1	Rezerv Değeri: 0
6. Sıra	7. Sıra	8. Sıra	9. Sıra	10. Sıra	Platform Maliyeti : 1	Platform Sayısı: 0
11. Sıra	12. Sıra	13. Sıra	14. Sıra	15. Sıra		Sondaj Maliyeti: 0
16. Sıra	17. Sıra	18. Sıra	19. Sıra	20. Sıra	Hesapla	Platform Maliyeti: 0
						Toplam Maliyet: 0
						Toplam Kar: 0

Parser Algoritması

CURL ile verilen web sitesi URL'si üzerinden koordinatları aldık. Daha sonra bu koordinatları bir diziye atadık. Bu dizi üzerinde while döngüsü ile ilerleyerek satır sonunu ifade eden 'F' karakterine kadar olan kısmı satır 1 ile ilgili diziye atadık. Dizi üzerinde ilerleyerek 'F' karakteri ile her karşılaştığımızda ilgili satırın dizisine atama gerçekleştirdik. Satırlardaki her bir şeklin başlangıç ve bitiş koordinatları aynıdır bundan yararlanarak satırdaki şekil sayısını ve bu şekillerin koordinatlarını bulduk. SDL kütüphanesi ile kullanılabilecek hale getirdik.

Ray Crossing Algoritması

Verilen şeklin minimum,maksimum X ve Y koordinatları tespit edilir. Her nokta için hayali bir ışın oluşturulur. Oluşturulan ışın şekli tek sayıda kesiyor ise bu nokta şeklin içinde olduğunu ifade eder. Bu yöntem ile tespit ettiğimiz noktalar birim karelerimizin başlangıç koordinatlarını ifade eder. Arayüzün boyutuna göre belirlediğimiz on piksellik birim kareleri ekrana çizdirir.

Optimal Karelere Ayırma Algoritması

Şeklin köşe noktasındaki kareden başlayarak etrafındaki diğer kareleri şeklinde içinde olup olmadığını tespit eder. Eğer şeklin içinde oluşturulabilecek daha büyük boyutta bir kare varsa kare büyük kareyi çizdirir ve komşu kare için de aynı işlemi gerçekleştirir. Optimal kare oluştururken hedeflenen amaç kârı maksimum maliyeti minimuma indirmektir.

Rezerv değeri,platform sayısı,sondaj maliyeti,platform maliyeti,toplam maliyet,toplam kâr adını verdiğimiz kutulardaki bu değerlerin hesaplanması için aşağıdaki formüller kullanılmıştır.

Formüller:

Rezerv Değeri = Alan x 10

Platform Sayısı = Kare Sayısı

Toplam Sondaj Maliyeti = Birim Sondaj Maliyeti x Alan

Toplam Platform Maliyeti = Birim Platform Maliyeti x Platform Sayısı

Toplam Maliyet = Toplam Platform Maliyeti + Toplam Sondaj Maliyeti

Toplam Kâr = Rezerv Değeri – Toplam Maliyet

IV. SONUÇ

Kullandığımız ideye göre uygun kütüphaneler yüklemeyi ve yüklediğimiz bu kütüphaneleri ideyekurmayı deneyimledik.

CURL kütüphanesi ile bir web sitesi URL'sinden veri çekmeyi deneyimledik.

SDL kütüphanesi ile kullanıcı dostu arayüzü oluşturmayı ve bu arayüzde şekiller çizdirip butonlar,kutular oluşturmayı deneyimledik.

SDL_TTF ile oluşturduğumuz bu arayüz üzerine yazılar yazdırmayı deneyimledik.

Gerçek hayat problemlerine çözüm üretme becerimizi geliştirdik.

Görüntü işleme ve modellemede grafiklerin daha kolay işlenebilmesi için kullanılan ray crossing algoritmasını projemizde deneyimledik.

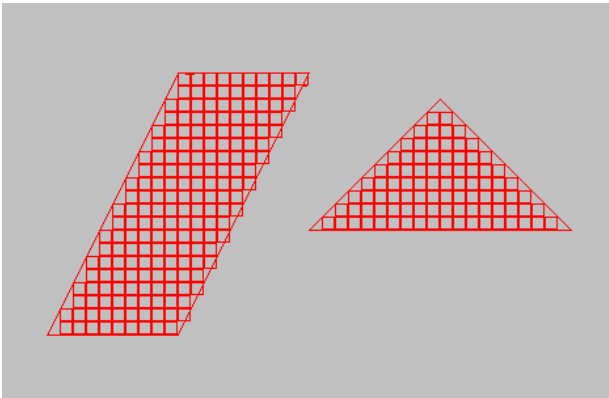
Takım çalışması ile iş bölümü yaptık.

V. DENEYSEL SONUÇLAR

CURL kütüphanesi ile kendimizin oluşturduğu URL üzerinden verileri çekme işlemi:

```
18(5,5)(13,12)(8,17)(1,10)(5,5)F
28(20,20)(30,20)(20,40)(10,40)(20,20)(40,22)(50,32)(30,32)(40,22)F
38(15,15)(22,15)(25,20)(20,25)(15,20)(15,15)F
48(15,15)(22,15)(25,20)(20,25)(15,20)(15,15)(5,5)(13,5)(15,10)(10,15)(5,10)(5,5)F
58(25,25)(30,25)(30,30)(25,30)(5,5)(10,5)(15,10)(15,15)(10,20)(5,20)(0,15)(0,10)(5,5)F
68(5,5)(13,12)(8,17)(1,10)(5,5)(50,50)(60,50)(65,70)(45,70)(50,50)F
78(3,5)(10,12)(3,15)(3,7)(3,5)F
88(3,5)(11,13)(5,16)(2,9)(3,5)F
98(110,110)(110,130)(130,130)(130,110)(110,110)F
108(20,20)(30,20)(20,40)(10,40)(20,20)(40,22)(50,32)(30,32)(40,22)F
118(4,7)(10,12)(22,20)(7,30)(4,7)F
128(20,20)(30,20)(20,40)(10,40)(20,20)(40,22)(50,32)(30,32)(40,22)(5,5)(13,12)(8,17)(1,10)(5,5)F
138(4,3)(4,9)(8,9)(13,5)(11,2)(4,3)(10,16)(21,18)(12,29)(10,16)(28,14)(34,10)(34,25)(28,28)(28,14)F
```

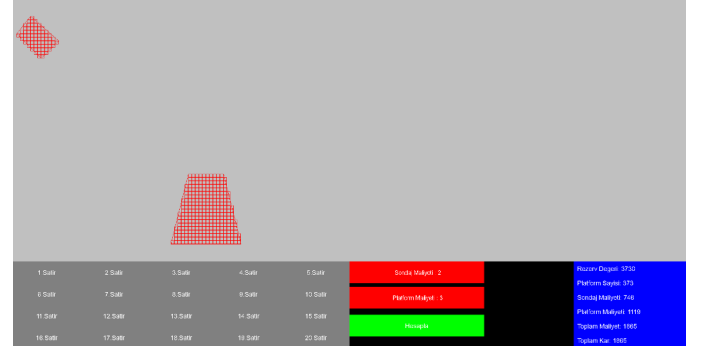
CURL kütüphanesi ile alınan veriyi SDL kütüphanesini kullanarak ekrana şekil veya şekiller çizdirme işlemi:



Birim sondaj maliyeti,birim platform maliyeti,rezerv değeri,platform sayısı,sondaj maliyeti,platform maliyeti,toplam maliyet,toplam kâr değerlerinin hesaplanarak ekranda gösterilme işlemi:

Sondaj Maliyeti : 2	Rezerv Degeri: 730
Platform Maliyeti : 3	Platform Sayisi: 73
Hesapla	Sondaj Maliyeti: 146
	Platform Maliyeti: 219
	Toplam Maliyet: 365
	Toplam Kar: 385

Örnek veriler ile hesaplamalar yapıldıktan sonra arayüzün son halinin gösterilmesi:



Referanslar:

<https://wiki.libsdl.org/SDL2/FrontPage>

<https://curl.se/docs/>

https://en.wikipedia.org/wiki/Point_in_polygon

<https://www.youtube.com/watch?v=RSXM9bgqxJM>

VI. YALANCI KOD

Parser Algoritması

- 1-CURL kullanarak URL'ye eriş.
- 2-URL'deki verileri çek.
- 3-Çektiğin verileri diziye ata.
- 4-While döngüsü ile dizi içinde 'F' karakterini ara.
- 5-'F' karakterinden önceki koordinatları satı 1 dizisine at.
- 6-Bu işlemi diğer satırlar için tekrarla.
- 7-Satırlar içindeki tekrara eden koordinatları bul.
- 8-Bu koordinatlar arasındaki veriyi başka bir diziye ata.

Ray Crossing Algoritması

- 1-Min X min Y max X max Y değerlerini bul.
- 2-İç içe for döngüsü kurarak aralık içerisindeki tüm noktaları tara.
- 3-Noktadan doğrusal bir ışın oluştur.
- 4-Işının kestiği kenar sayısını bul.
- 5-Kesişim sayısı tek ise kare çizdir.

Optimal Boyutlarda Karelere Ayırma Algoritması

- 1-Şeklin bir tane köşesindeki kareyi tespit et.
- 2-Karenin etrafındaki diğer kareler şeklin içinde mi diye kontrol et.
- 3-Eğer birim kareler şeklin içinde daha büyük kare oluşturabiliyor ise büyük kareyi çizdir.
- 4-Aynı kontrolü komşu kare için yap.