

Bilgisayar Destekli Rezerv Arama için C Tabanlı Program

Batın Erke ÖZTÜRK
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli/Türkiye
batinerkeozturk@gmail.com

Mehmet Ali DEMİR
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli/Türkiye
mehmetalidemir1370@gmail.com

— Bu rapor 'Rezerv Arama' adlı Programlama Laboratuvarı dersinin 1. Projesini açıklamak ve sunumunu gerçekleştirmek amacıyla oluşturulmuştur. Rapor projenin tanımı, yapım aşaması, kullanılan fonksiyonlar ve yöntemler bulunmaktadır.

I. ÖZET

Denizlerde doğal kaynak arama ve çıkarma uygulamaları yapan bir şirket rezerv alanlarını en optimal karelere bölerek doğal kaynak çıkarmak istemekteydi. Bunun için belirli kuralları olan bir yazılıma ihtiyaçları vardı. Biz de bunun için bir yazılım geliştirdik. Öncelikle verilen URL'den rezerv alanlarının koordinatlarını alınacak sonra kullanıcının konsol ekranından girdiği belirli bilgilere göre pencerede rezerv alanlarının bölünmüş halini görmesini ve konsolda kar-zarar durumunu görmesini sağladık.

A. PROJE AŞAMALARI VE ÇÖZÜMLERİ

Bu proje 2. aşamadan oluşmaktadır. Projenin ilk aşaması verilen URL'den koordinatları almak ve sonucunda koordinat noktalarını birleştirerek 2 boyutlu şekil oluşturmaktır. 2. aşama ise sondaj ve kaynak çıkarma yapmaktır. Bunun için en optimal boyutlarda ve biçimlerde kareler kullanmak gerekir. Bu aşama sırasında belirli şartlar göz önünde olmalıdır.

B. 1. AŞAMA

1.a) Verilecek URL linkinde yer alan web sayfasında bulunan koordinatlar (x,y) şeklinde okunacaktır.

1.b) Okunan bu veriler verilen sırayla doğrular çizilerek iki boyutlu kapalı alanlı şekiller oluşturulacaktır.

1.c) Çizilen şeklin yüzey alan değeri hesaplanacaktır. (Yüzey alanı hesaplanırken şeklin kenarlarının içinde olduğu ve değdiği tüm birim kareler alınacaktır).

1.d) Yüzey alan değeri 10 ile çarpılarak rezerv değeri elde edilecektir.

C. 2. AŞAMA

2.a) Bölünen tüm karesel alanlarda sondaj faaliyeti yapılacaktır.

2.b) Toplam sondaj maliyetinin en az tutulması istenmektedir.

2.c) Toplam sondaj maliyeti karesel alan sayısının sondaj maliyetiyle çarpımından elde edilecektir. (Sondaj maliyeti 1-10 arasında olmalıdır.)

2.d) Kaynakların depolanması için platformlar kurulacaktır ve her karesel alanda bir platform olması istenmektedir.

2.e) Toplam platform maliyetinin en az olması istenmektedir bunun için mümkün olduğunca az platform kullanılmalıdır.

2.f) Sondaj maliyeti ve platform maliyeti konsoldan yani kullanıcıdan alınmalıdır.

2.g) Kapalı şekiller karesel alanlara bölünecektir. Bölünen karesel alanların boyutları 1x1, 2x2, 4x4, 8x8, 16x16'lık olmalıdır.

2.h) Ayrıca her karesel alanın komşusu kendinden bir büyük veya bir küçük karesel alan olmalıdır yani 16x16'lık karesel alanın yanında 2x2'lik karesel alan olamaz. (Köşe noktaları alan sınırı değildir.)

2.i) Bir karesel alanın içine aldığı birim kareyi başka karesel alan içine almayacaktır.

2.j) Son olarak kullanıcıya kar-zarar durumu gösterilmelidir.

II. AŞAMA ÇÖZÜMLERİ

1. Aşama Çözümü

Curl kütüphanesini başlatmak için `curl_global_init()` kullanıyoruz ve `CURL_GLOBAL_DEFAULT` ise varsayılan başlatmayı kullanıyor. Sonrasında curl işlemi oluşturuyoruz. Curl işlemimiz başarılı olursa gerekenleri yapmaya devam ediyoruz. `curl_easy_setopt` işlemini kullanarak curl işlemine URL'yi belirten bir seçenek olarak `CURLOPT_URL` yi ekliyoruz.

```
curl_easy_setopt(curl, CURLOPT_URL,
"http://bilgisayar.kocaeli.edu.tr/prolab1/prolab1.txt");
```

Burada ise `CURLOPT_WRITEFUNCTION` ile veriyi almak için kullanılacak işlevi belirtiyoruz.

Veri alma işlemi başarılı olduysa veriyi `VeriAlmaCallBack` fonksiyonumuzla tanımladığımız bir char veri isimli dizimize atıyoruz.

Sonra veri isimli dizimizi satırlara bölmemiz gerektiği için (`\n`)'i kullanarak satırlara bölüp var olan her satırı satırlar dizilerimize teker teker atıyoruz.

Satırları ayırma işlemimizden sonra kullanıcıdan çizdirmek istediği satır sayısını, platform maliyetini ve sondaj maliyetini konsoldan girmesini istiyoruz.

```
Hangi satiri çizdirmek istersin? (1-20): 2
Birim sondaj maliyetini giriniz : 1.3
Birim platform maliyetini giriniz : 2.4
```

Sondaj maliyeti 1-10 arasında ve platform maliyeti 0'dan büyük değilse ekrana hata veren bir kod parçası yazıyoruz ve kod sonlanıyor. Eğer istenen durumlarda kod devam ediyor.

```
Birim sondaj maliyetini giriniz : 4
Birim platform maliyetini giriniz : -2

Yanlis platform maliyeti girdiniz !
Birim platform maliyetini giriniz : 2.4
Yanlis sondaj maliyeti girdiniz !
```

Bu aşamaya geldikten sonra X ve Y koordinatlarımızı ayrı ayrı tutacak integer dizileri ve koordinat sayımızı tutacak bir (`koordinat_sayisi`) değişkenini tanımlıyoruz.

Koordinatları char dizisinin içinden alıp integer'a çevirdikten sonra bunları dizilerimize koyuyoruz.

Şekillerimiz pencerede çok küçük gözüktüğünden ve kullanıcının daha iyi görmesi açısından koordinatlarımızı 10 ile çarpıyoruz (Ekran yazdırırken ve kullanıcıya gösterilirken asıl koordinatlar gösterilecektir.)

Şekilleri ayırmak için koordinat sayısını kullanmamız gerekiyor ama bize asıl koordinat sayısının ilerde lazım olmasından dolayı (`koordinatsayisi`) isminde benzer bir integer değişkeni tanımlıyoruz.

Seçilen satırdaki tüm koordinatları ekrana yazdırıyoruz.

```
Koordinatlar:
X: 20, Y: 20
X: 30, Y: 20
X: 20, Y: 40
X: 10, Y: 40
X: 20, Y: 20
X: 40, Y: 22
X: 50, Y: 32
X: 30, Y: 32
X: 40, Y: 22
```

Bize lazım olan diğer diziler[], diziiki[], diziuc[] dizilerini ; şekillerin koordinatlarını kontrol edeceğimiz `sayacbir`, `sayaciki`, `sayacuc` tam sayılarını ; `toplamsondajmaliyeti`, `toplamlatformmaliyeti`, `toplammaliyet`, `karzarardurumu` ondalıklı sayılarını; ve kareleri saymamızda yardımcı olacak `birlik` sayısını, `ikilik` sayısını, `dortluk` sayısını, `sekizlik` sayısını, `onaltilik` sayısını isimli tamsayı değişkenlerimizi tanımlıyoruz.

Birinci şeklin ilk koordinatlarını elimizle dizinin ilk iki elemanına atıyoruz.

Aynı x ve y ikilisine gelene kadar şeklimizin koordinatlarını dizibir dizimize alıyoruz.

Eğer koordinat sayımız başlangıçtaki 2 sayısından fazlaysa bir şekil olduğunu anlıyor ve onu Şekil 1 ismiyle konsola yazdırıyoruz.

```
1. Sekil Koordinatlari : 20 20 30 20 20 40 10 40 20 20
```

Eğer ikinci bir şekil varsa `koordinatsayisi` değişkenimiz ile kontrol ediyor ve almaya başlıyoruz.

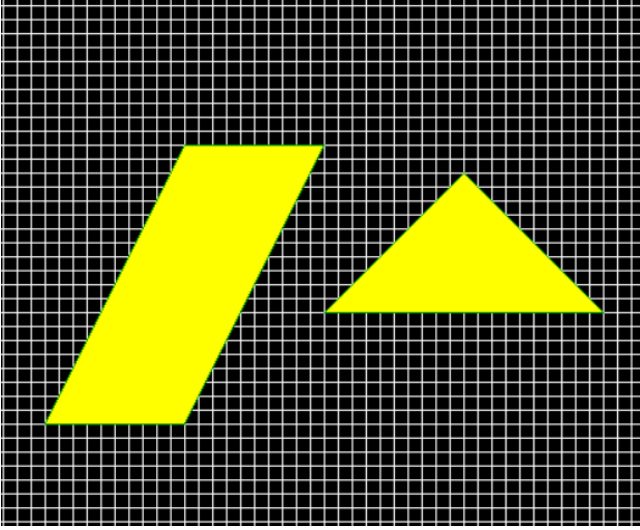
İkinci şekil varsa koordinatları ekrana yazdırıyoruz.

```
2. Sekil Koordinatlari : 40 22 50 32 30 32 40 22
```

Üçüncü şekil varsa kontrol edip ekrana yazdırıyoruz.

İnitwindow ile şekillerimizi çizeceğimiz pencereyi açıyoruz ve arkaya (x*10, y*10)'luk karelerimizi yerleştirerek birim kareli pencere oluşturuyoruz.

Sonrasında şekillerimizi içi dolu bir şekilde 2 boyutlu olarak penceremizde çizdiriyoruz ve ilk aşamamız bitiyor.



2. Aşama Çözümü

Klavyeden bir tuşa basınca çizimimiz devam ediyor.

Devam edilen çizimde sarı renkle boyadığımız şekillerin içlerini ve şekillerin kenarlarının değdiği birim kareleri açık kırmızı kullanarak 1x1' lik karelere ayırıyoruz.

Şekillerin toplam alanını ve rezerv değerini (toplam alan * 10) konsola yazdırıyoruz.

```
TOPLAM ALAN : 330 BirimKare
Rezerv Degeri : 3300
```

Sonrasında eğer uygunsa 1x1 'lik karelerimizi 2x2 'lik olarak güncelliyoruz. (2x2 'lik karelerimizin rengi yeşildir).

2x2 'lik karelerimizi yerleştirdikten sonra şekillerimiz 4x4 'lük kareler için uygunsa 4x4 lük kareleri ekliyoruz. 1x1 'lik karelerin 4x4 'lük karelere komşu olmasını istemediğimiz için kontroller ederek ekliyoruz. (4x4 'lük karelerimizin rengi magentadır).

Kodumuzun devamında 2x2 'lik kareleri 8x8 'lik karelere komşu olmayacak şekilde kontrol ederek 8x8 'lik karelerimizi ekliyoruz. (8x8 'lik karelerimizin rengi açık mavidir).

En sonunda 4x4 'lük karelerinde 16x16 'lık karelere komşu olmasını istemediğimiz için kontrol ederek 16x16 'lık karelerimizi ekliyoruz. (16x16 'lık karelerimizin rengi kahverengidir.)



Burada artık gereken işlemlerimizi yapıyoruz ve kullanıcıya toplam sondaj maliyeti, toplam platform sayısı, toplam platform maliyeti, toplam maliyet, kar-zarar durumunu kısacası tüm konsolu gösteriyoruz.

```
Veri :
1B(5,5)(13,12)(8,17)(1,10)(5,5)F
2B(20,20)(30,20)(20,40)(10,40)(20,20)(40,22)(50,32)(30,32)(40,22)F
Hangi satiri çizdirmek istersin? (1-20): 2
Birim sondaj maliyetini giriniz : 1.3
Birim platform maliyetini giriniz : 2.4
2.Satir: 2B(20,20)(30,20)(20,40)(10,40)(20,20)(40,22)(50,32)(30,32)(40,22)F
Koordinatlar:
X: 20, Y: 20
X: 30, Y: 20
X: 20, Y: 40
X: 10, Y: 40
X: 20, Y: 20
X: 40, Y: 22
X: 50, Y: 32
X: 30, Y: 32
X: 40, Y: 22
1. Sekil Koordinatlari : 20 20 30 20 20 40 10 40 20 20
2. Sekil Koordinatlari : 40 22 50 32 30 32 40 22
TOPLAM ALAN : 330 BirimKare
Rezerv Degeri : 3300
Toplam Sondaj Maliyeti : 429.00
Toplam Platform Sayisi : 96
Toplam Platform Maliyeti : 230.40
Toplam Maliyet : 659.40
Kar - Zarar Durumu : 2640.60
```

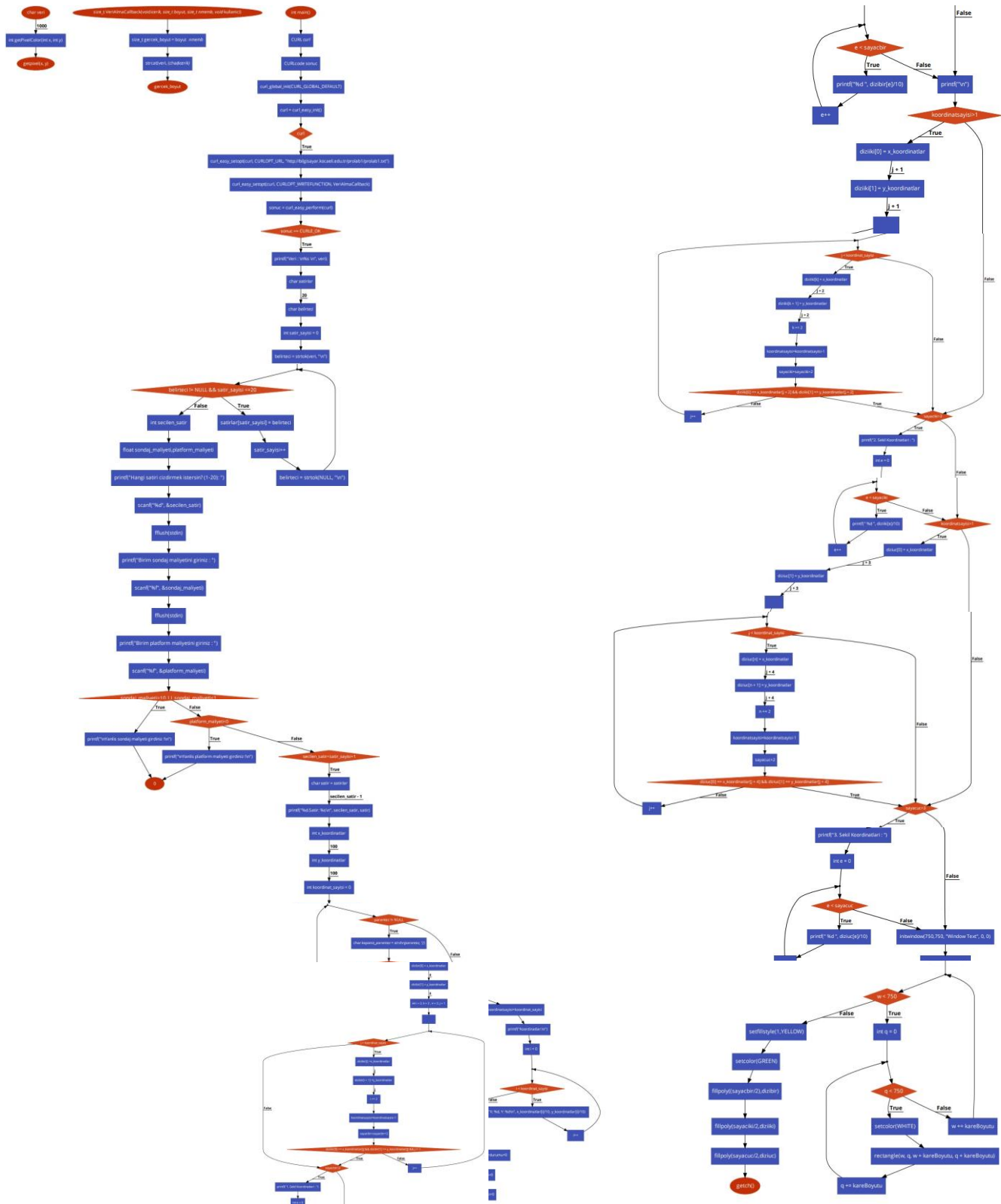
Eğer satır sayısı 1-20 arasında bir sayı değil de başka bir sayı girilirse satır numarasında hata veriyoruz.

Curl çalışmazsa hata veriyoruz.

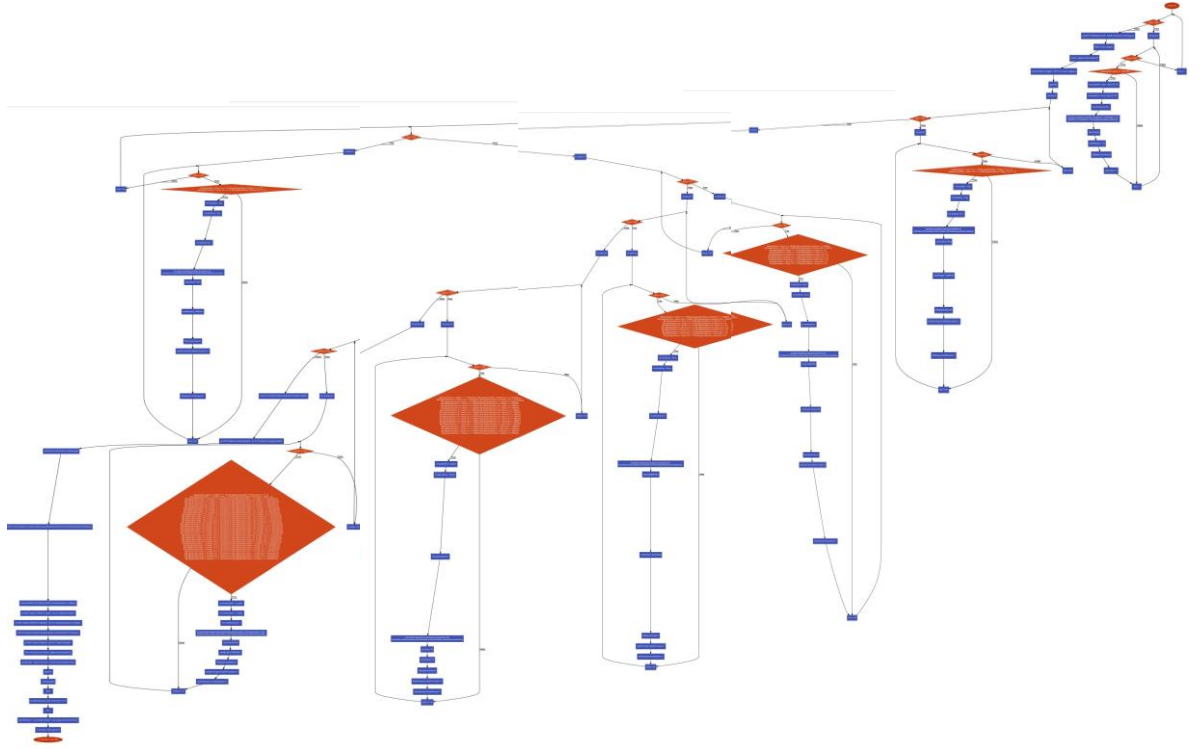
İşlemlerin hepsi tamamlanınca kaynakları serbest bırakıyoruz.

Kütüphaneyi temizledik ve kaynakları serbest bıraktık.

1. AŞAMA AKIŞ DİYAGRAMI



2. AŞAMA AKIŞ DİYAGRAMI



KAYNAKLAR

1. <https://www.youtube.com/watch?v=7CRf4nZ5CRw>
2. [libcurl example - simple.c](#)
3. <https://www.youtube.com/watch?v=hgtC-gu08zQ&list=PLD5D5Hj95BCFid63gy2VtVBXJZQXuw12R&index=14>
4. [C Programlama Dili WinBGIm Renkler | C Kaynak](#)
5. <https://www.youtube.com/watch?v=Lhbu6KpTdCc>