Kocaeli Üniversitesi
Doğal Kaynak Arama ve Çıkarma Projesi

Berk GÜVEN
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli,Türkiye
berkconfidence@gmail.com

Bu rapor, 'Doğal Kaynak Arama ve Çıkarma' adlı Programlama 1 dersinin 1.projesini açıklamak ve sunumunu gerçekleştirmek için yapılmıştır. Raporda projenin özeti, çözümü, yöntemi ve akış şeması bulunmaktadır.

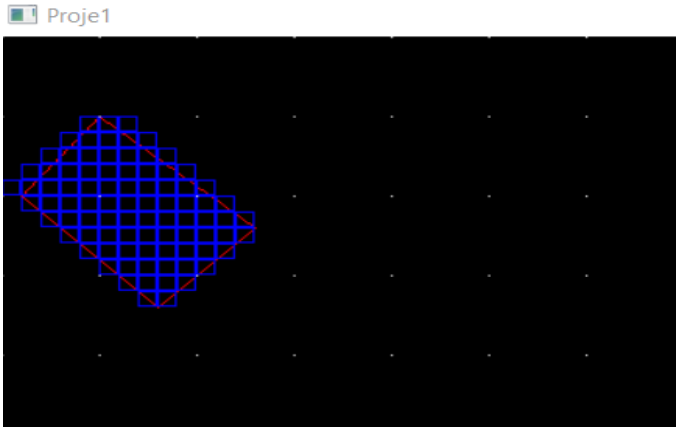
I. ÖZET

Bu proje 2 aşamadan oluşmaktadır. 1. aşaması kaynak arama şirketinin sismik araştırma ayağını oluşturmaktadır. URL'ye bağlı txt dosyasından alınacak koordinat noktalarının birleştirilmesiyle kapalı bir şekil veya şekiller oluşturulacaktır. Bu şeklin alanına bağlı olarak rezerv değeri hesaplanacaktır. Projenin 2. aşaması, 1. aşamada yüzey alanları üzerinden rezerv değer miktarı tespit edilen bölgelerde kaynak arama şirketinin sondaj ve kaynak çıkarma ayağını oluşturmaktadır. İlk aşamada çizilen rezerv alanlarının en optimal biçimde belirli boyutlardaki düzgün karesel parçalara bölünmesi gerekmektedir. En son maliyetlere bağlı olarak kâr miktarı hesaplanıp kullanıcıya gösterilmelidir.

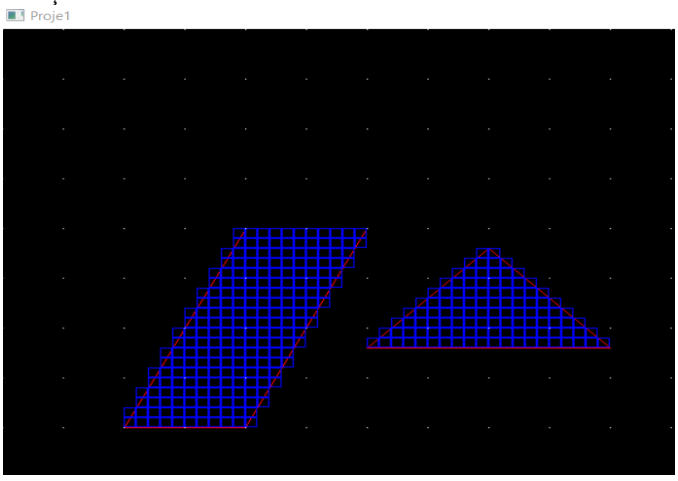
II. ÇÖZÜM

Projenin 1. aşamasını çözmek için öncelikle, sistemde paylaşılan url'den koordinatları çekmek için Curl kütüphanesini kuruyoruz. Kütüphaneyi kurduktan sonra Curl'ün fonksiyonlarını kullanarak url'deki verileri depo adlı değişkende saklıyoruz ve değişkende daha kolay değişiklik yapmak ptr adlı pointer'a eşitliyoruz. Daha sonra koordinatları pointerdan x y dizilerine aktarıyoruz. Aldığımız koordinatları daha sonra alanhesapla fonksiyonuna gönderip çokgenlerin alanını bulma formülüyle oluşan şeklin alanını yani rezerv değerini hesaplayıp geri döndürüyoruz.

Program alanı hesapladıktan sonra sıra çokgeni çizdirmeye geçiyor. Bu sefer ise x y dizilerini asama1 adlı fonksiyona gönderiyoruz. Bu fonksiyonda işlem yapabilmek, arayüz oluşturmak için SDL kütüphanesini kuruyoruz. Kütüphane kurulduktan sonra ve değişkenleri bu fonksiyona yolladıktan sonra ilk aşamada istenilen, çokgeni çizdirmek ve birim karelere ayırmak için öncelikle SDL'in fonksiyonları ile önce pencere daha sonra ise çizim yüzeyi oluşturuyoruz. While döngüsü ile ekranı escape tuşu veya kapatma düğmesine basılana kadar açık tutuyoruz. Döngü içerisinde ise bir dizi değişken ve for döngüleriyle çizimi yapıyoruz. Böylece projenin ilk aşamasını tamamlamış oluyoruz.



1. Aşama ikinci satır:



1.aşama bittikten sonra bu sefer x y dizilerini ve bazı değişkenleri asama2 adlı fonksiyona gönderiyoruz. Aynı SDL fonksiyonlarını da burada uyguluyoruz. Yaptığım algoritma ile de bu sefer sondaj maliyetine göre optimize 1,2,4 veya 8 birimlik karelerle platformları yerleştiriyoruz. En son sondaj maliyeti ve platform maliyetleri ve rezerv değerleriyle son işlemleri yapıyoruz.

III. YÖNTEM

Kullanıcıdan, url'den gelen satılardan scanf fonksiyonu ile seçim yapmasını istiyoruz ve bunu satirnumarasi adlı değişkende saklıyoruz. String kütüphanesinden olan strtok adlı fonksiyon ile satır numarasına göre, istenilen yeri ikiye ayırmak için kullanıyoruz. Bu ayırdığımız kısmı satir adlı pointer'da saklıyoruz. Daha sonra koordinatları daha kolay x ve y dizisine aktarmak için satir adlı fonksiyonu da strcpy adlı fonksiyonu kullanarak dizi adlı diziye eşitliyoruz.

- For döngüsü ile dizi[i]!= '\0' değerine kadar döndürüyoruz, if kontrolüyle dizi[i] == '(' değerine eşit olunca son if kontrolü ile sscanf(dizi + i, "(%d,%d)", &x[sayac], &y[sayac]) == 2) dizinin ikili elemanlarını x ve y'ye aktarıyoruz.

Daha sonra sayaç değişkeni ile x ve y dizilerindeki elemanları x1 ve y1 dizisine for döngüsüyle aktarıyoruz. Url'den verilen koordinatlarda başlangıç ve bitiş değerleri aynı odluğu için:

- If ile (i!=0 && x[0]==x[i] && y[0]==y[i]) ilk çokgenin nerede biteceğini kontrol ediyoruz ve bir if ile daha ondan sonraki değerlerin null olup olmadığını kontrol ettim. Eğer null ise bir adet çokken değil ise 2 adet çokgen olduğunu varsaydım.
- Null değilse while döngüsü ile kalan elemanları x2 ve y2 fonksiyonlarına aktardım ve bir çokgen varsa yenidevam değişkenini 0'a daha fazla çokgen varsa yenidevam değişkenini 2'ye eşitledim

Değerlerimizi aktardıktan sonra yenidevam değişkeninin değerine göre çizdirme aşamasına geçmeden önce, oluşan şeklimizin alanını hesaplamak amacıyla alanhesapla adlı fonksiyonu kullanıyoruz. Buraya x ve y dizilerini gönderip matematiksel alan hesaplama formülü ile çokgenin rezerv değerini hesaplıyoruz. Return ederek alan adlı değişkende bu değeri saklıyoruz ve yazdırıyoruz. (Bütün bu durumların aynısını satirnumarasi değeri 2 olursa da yapıyoruz) Alan hesaplama da bittikten sonra sıra asama1 fonksiyonuna geliyor.

Asama1 fonksiyonuna çokgen sayısına bağlı olarak 4 adet dizi(x1,y1,x2,y2) ve çokgen sayısını ifade eden bir değişken gönderiyoruz. Fonksiyonda ilk olarak Sdl'in window ve render fonksiyonlarını pencere açma ve çizim yüzeyi oluşturma amacıyla kullanıyoruz. Daha sonra yapılacak olayları kontrol etmesi için event fonksiyonunu kullanıyoruz ve while döngüsünde devam==1 olduğu sürece çizime başlıyoruz.

- While döngüsünü istediğimiz zaman kapatabilmek için event.type == SDL_QUIT veya event.key.keysym.sym == SDLK_ESCAPE kontrolünü yaparak bu şartlar için devam = 0 durumunu yazdığımızda while döngüsünü durduruyoruz.
- SDL_SetRenderDrawColor(renderer, 0, 0, 0, 255); SDL_RenderClear(renderer); fonksiyonları ile rengimizi seçiyoruz ve ekranı daha önceden çizim kalmışsa diye temizliyoruz.

Asama1'e gönderdiğimiz son değişkende kaç tane çokgen çizdireceğimize bağlı olarak if kontrolü ile çizime başlıyoruz. Bizden istenen ilk aşamada çokgeni çizdirmek ve birim karelere ayırmak olduğu için birim karelere ayırma algoritması tasarladım. Bu algoritmada çokgenin bir köşesi ile diğer köşesini karşılaştırıp bu iki köşenin arasında kalan kenarın koordinat sistemindeki durumunu hesaplıyoruz. Bu 4 ayrı durumdan bir tanesi olduğu zaman ona özel olarak o iki köşe arasında satır satır birim kare yerleştirmeye başlıyor. For döngüsü ile de koordinat sistemindeymiş gibi gözükmesi için 50 birimlik aralıklarla noktalar koydum ve böylelikle bizden istenen 1.aşamayı tamamlamış oluyoruz.

Asama2 fonksiyonuna geçtiğimizde ise daha zorlu bir aşama olan optimizasyon problemi bizi karşılıyor. Bu fonksiyona da asama1'deki değerlerin haricinde ekstra olarak kâr miktarını hesaplamak için alan değişkeninde sakladığımız rezerv değerini gönderiyoruz. Bu fonksiyonda ise sondaj ve platform değişkenlerini kullanarak kullanıcıdan maliyet değerlerini alıyoruz. Ek olarak sondaj yapılan alan için birim, konulacak platform sayısını temsil etmek için de adet değişkeni tanımlıyoruz. Aşama 1'deki gibi yine devam değişkeni kullanarak while döngüsü içinde çizimimize başlıyoruz. Bu aşamada yapılması gereken problem için genel bir çözüm sunmak ama iki haftalık uğraş sonucu doğru çözümü yapamadığım için bize verilen url'deki koordinatlara göre probleme özgü bir çözüm yapmak zorunda kaldım. Bu yazdığım algoritmanın mantığı çokgenin iki ucunu xmin, xmax, ymin, ymax değişkenlerinde saklanıyor ve 3 duruma göre çalışıyor.

- Bu durumlar sondaj>=7, sondaj>3 && sondaj<7, sondaj<=3 'dür.

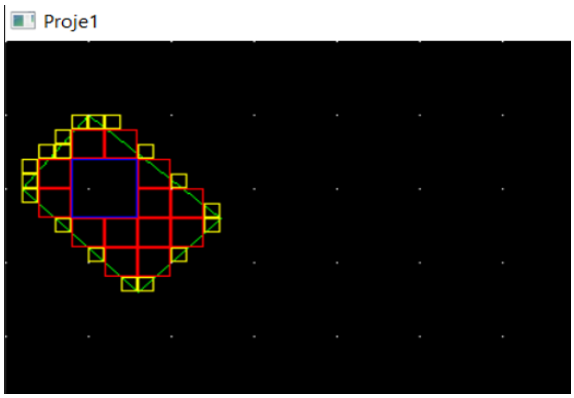
Eğer kullanıcıdan alınan değere göre sondaj>=7 durumu gerçekleşirse platform boyutları daha küçük ve çokgenin dışına çıkmayacak şekilde, eğer sondaj>3 && sondaj<7 olursa platform boyutları diğer duruma nazaran daha büyük ve çokgenin dışında daha az taşıyor. Son durum olan sondaj<=3 için platform boyutları daha maksimize ve biraz daha çokgenin dışına taşmış şekildedir. Bu durumlara göre yerleştirdiğimiz platformlar sonucunda adet sayısını artırıyoruz. Sondaj yapılan alanı bulmak için de konulan platformun boyutuna bağlı olarak birim kareleri hesaplıyoruz. Son aşamaya gelmeden önce ise:

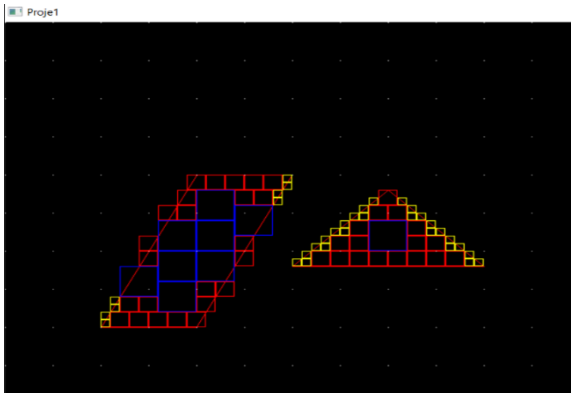
- sondajmaliyeti=birim*sondaj;
  platformmaliyeti=adet*platform;
  toplammaliyet=platformmaliyeti+sondajmaliyeti;

Maliyetleri hesaplıyoruz. Aynı durumu 2.satır seçilirse de uygulanıyor ve ek olarak 2. bir çokgen olduğu için birim2 ve adet2 ve sondajmaliyeti2, platformmaliyeti2 değişkenleri de işin içine giriyor.

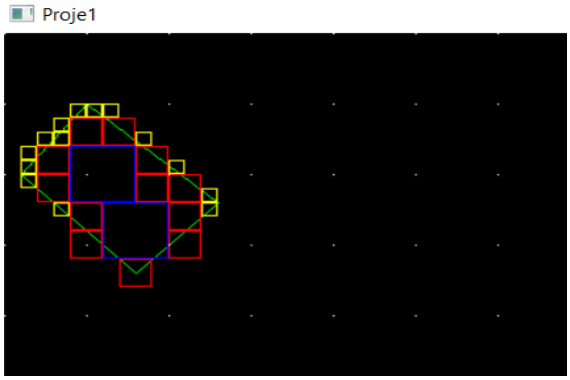
Son aşamaya geldiğimizde ise tüm maliyet ve durumları hesapladıktan sonra sıra bu değerleri kullanıcıya göstermek kalıyor. Kullanıcıya sırasıyla: adet, birim, platformmaliyeti, sondajmaliyeti, toplammaliyet ve kâr miktarını gösteriyoruz. Böylelikle 2. Aşama da bitmiş oluyor.

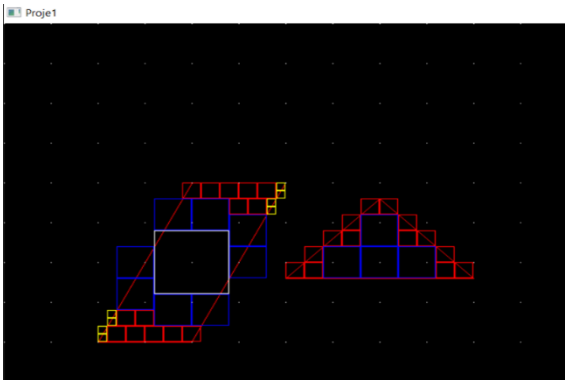
- Sondaj>=7 için ekran çıktısı



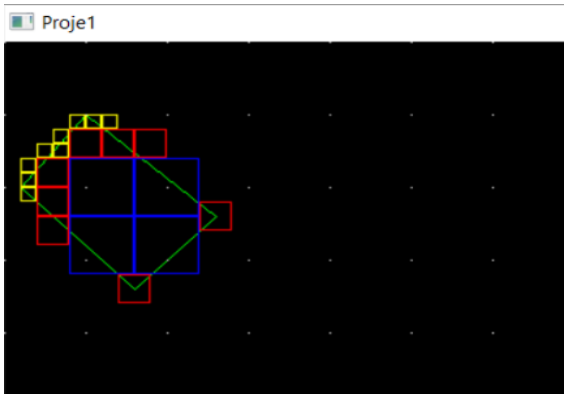


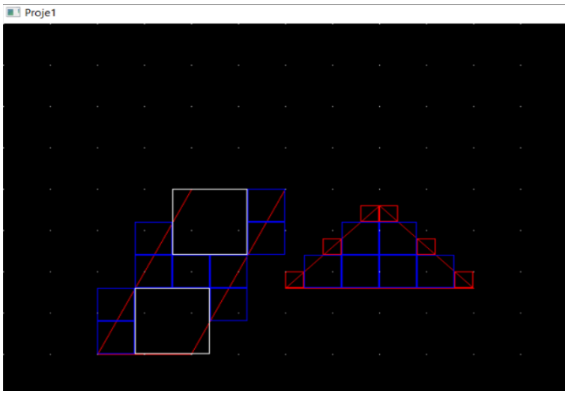
- Sondaj>3 && sondaj<7 için ekran çıktısı





- Sondaj<=3

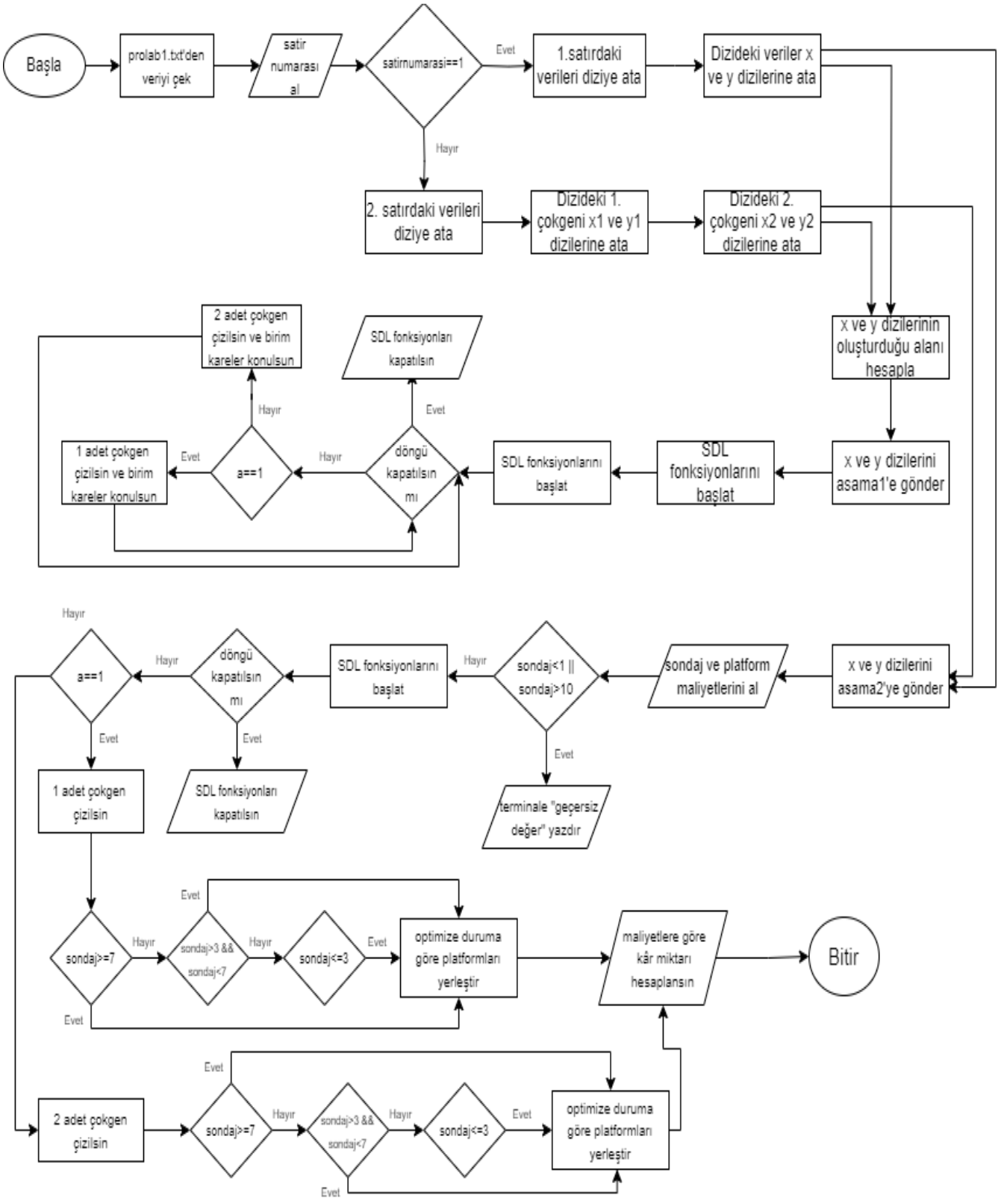




- Terminal

```
Seç C:\CodeBlocks\deneme\gui\bin\Debug\gui.exe
KORDINATLARIMIZ
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

Cizimi yapilacak satiri seciniz (1 ya da 2): 1
Kaynak rezerv degeri: 690

Sondaj maliyetini giriniz (1 ile 10 arasi): 7
Platform maliyetini giriniz: 2

Toplam platform sayisi: 32
Toplam sondaj sayisi: 86

Toplam platform maliyeti: 64
Toplam sondaj maliyeti: 602

Toplam maliyet: 666
Toplam kar miktari: 24

Process returned 0 (0x0)   execution time : 85.842 s
Press any key to continue.
```

```
C:\CodeBlocks\deneme\gui\bin\Debug\gui.exe
KORDINATLARIMIZ
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

Cizimi yapilacak satiri seciniz (1 ya da 2): 2
Kaynak rezerv degeri: 4000

Sondaj maliyetini giriniz (1 ile 10 arasi): 7
Platform maliyetini giriniz: 5

Toplam platform sayisi: 84
Toplam sondaj sayisi: 342

Toplam platform maliyeti: 420
Toplam sondaj maliyeti: 2394

Toplam maliyet: 2814
Toplam kar miktari: 1186

Process returned 0 (0x0)   execution time : 38.584 s
Press any key to continue.
```


Başla
prolab1.txt'den veriyi çek
satir numarası al
satirnumarasi==1
Evet
1.satırdaki verileri diziye ata
Dizideki veriler x ve y dizilerine ata
Hayır
2. satırdaki verileri diziye ata
Dizideki 1. çokgeni x1 ve y1 dizilerine ata
Dizideki 2. çokgeni x2 ve y2 dizilerine ata
x ve y dizilerinin oluşturduğu alanı hesapla
x ve y dizilerini asama1'e gönder
SDL fonksiyonlarını başlat
SDL fonksiyonlarını başlat
döngü kapatılsın mı
Evet
Hayır
SDL fonksiyonları kapatılsın
a==1
Evet
Hayır
1 adet çokgen çizilsin ve birim kareler konulsun
2 adet çokgen çizilsin ve birim kareler konulsun
x ve y dizilerini asama2'ye gönder
sondaj ve platform maliyetlerini al
sondaj<1 || sondaj>10
Evet
terminale "geçersiz değer" yazdır
Hayır
SDL fonksiyonlarını başlat
döngü kapatılsın mı
Evet
SDL fonksiyonları kapatılsın
Hayır
a==1
Evet
1 adet çokgen çizilsin
sondaj>=7
Evet
Hayır
sondaj>3 && sondaj<7
Hayır
sondaj<=3
Evet
optimize duruma göre platformları yerleştir
maliyetlere göre kâr miktarı hesaplansın
Bitir
2 adet çokgen çizilsin
sondaj>=7
Evet
Hayır
sondaj>3 && sondaj<7
Hayır
sondaj<=3
Evet
optimize duruma göre platformları yerleştir