**T.C.**

**BALIKESİR ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

metin, daire, logo, ticari marka içeren bir resim

Açıklama otomatik olarak oluşturuldu

**2D DÜZGÜN DİKDÖRTGENSEL PAKETLEME PROBLEMİ**

**202113709014 OĞUZHAN YILMAZ**

**BMM4101 YAPAY ZEKA TEKNİKLERİ FİNAL ÖDEVİ**

**Danışman: Dr. Öğr. Üyesi Kadriye ERGÜN**

**BALIKESİR, 01-2024**

**İÇİNDEKİLER**

**1-**Çalışmanın tanımı ve amacı

**2-** Kullanılan program veya yazılımlar hakkında bilgi verilmesive gerekçeleri

**3-** Kesme ve Paketleme Problemleri hakkında kısa bir bilgi, kullanılan yöntemin akış diyagramı, kullanılan parametreler

**4-**Ödevde yaptığınız çalışma ve seçtiğiniz yöntemlerin sırası ve anlatımı

**5-**Sonuç

**6-**Kaynaklar

**1-**

20X20 cm boyutundaki plaka üzerine boyutları verilmiş dikdörtgen parçalar yerleştirmek ve yerleştirirken problemde de en iyi yerleşimi elde etmek. Amaç parçalar öyle yerleştirilsin ki atık malzemenin en az olmasıdır. Yerleştirilecek parçaların veri girişi kullanıcı tarafından arayüzle sağlanacak ve yine bir arayüz ile kullanıcıya çıktı verilecek .

**2-**

Projeyi yaparken Python programlama dili kullandım. Python okunabilir ve anlaşılmasının kolay olması, içinde çok fazla kütüphane barındırması ve bu kütüphanelerin Python’da kullanımının kolay olması. Arayüz tasarımı için tkinter kütüphanesinden yararlandım çünkü Python kolayca kullanılıyor. Dikdörtgenlerin plaka üzerine yerleşimi görselleştirmek için ise matplotlib kütüphanesi kullandım.

**3-**

Kesme problemleri, bir malzemenin belirli boyutlarda parçalara bölünerek problemlerin çözümünde kullanılmasını amaçlar.

Paketleme problemleri, belirli boyutlardaki alanlara verilen nesneleri en iyi şekilde yerleştirme problemidir.

**Akış Diyagramı:**

**diyagram, çizgi, plan, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**4-**

Algoritma dikdörtgen parçaları alanına göre büyükten küçüğe doğru sıralıyor . Ardından en büyük alana sahip parçayı yerleştiriyor. Ardından, diğer parçaları ilk yerleştirilen dikdörtgen içine yerleştirmek için arama yapar. Her parçalar için en uygun yerleşimi belirlemek amacıyla belirli bir bölge içinde çakışma olup olmadığını kontrol eder. Çakışan parçalar olmaması halinde, parçayı en uygun konumda yerleştirir ve plaka üzerinde arta kalan alanı bir skor yapısı kullanır. Bu skor yapısı plaka üzerinde arta kalan alanı en aza indirmeye çalışır. Algoritma her parçanın en uygun yerleşimini belirlemek için düzenli bir arama yapar, böylece parçalar arasın arta kalan alanı en aza indirger.

metin, ekran görüntüsü, yazılım içeren bir resim

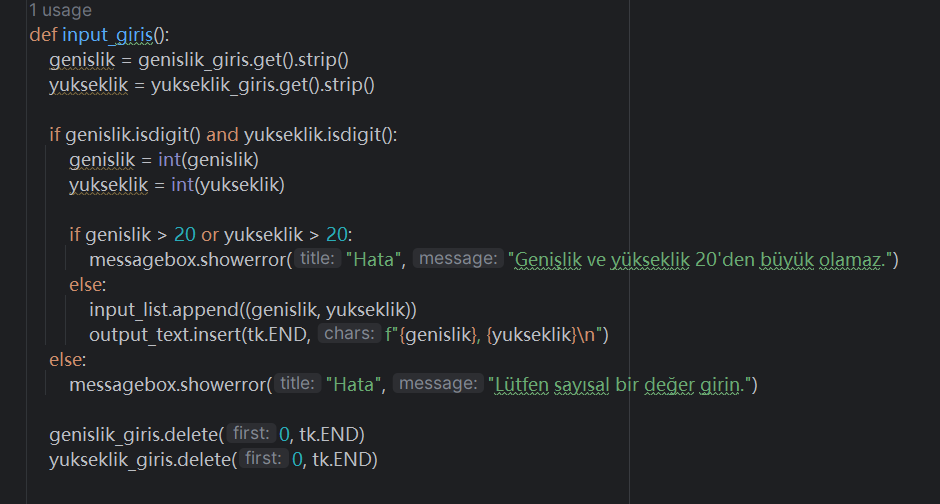
Açıklama otomatik olarak oluşturuldu

Dikdörtgenleri plaka üzerinde en uygun konuma yerleştirme işlemini yaptıktan sonra plaka üzerinde arta kalan alanı hesaplamak için alan\_hesaplama adında fonksiyon oluşturdum ve içerisinde işlemi gerçekleştirdim.

**metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

input\_giriş adlı fonksiyon ile genişlik ve yükseklik değişkenleri alınır. Bu değişkenler kullanıcıdan aldığımız verileri temsil eder. isdigit(**)** metodu ile genişlik ve yükseklik değerleri sayısal ise bu değerlerin integera dönüştürür. Eğer genişlik ve yükseklik değerleri geçerliyse, bu değerler **input\_list** adlı bir liste içine eklenir. Aynı zamanda, çıktı olan **output\_text** içine bu değerler eklenir. Eğer kullanıcı sayısal bir değer girmemişse veya hatalı bir giriş yapmışsa, hata mesajı penceresi ile uyarı verir.

****

Sonuç adlı fonksiyonda sonuç ekranında verilecek bilgiler hazırlanır. Ve sonuç penceresinin düzenlemesi yapılır.

**metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

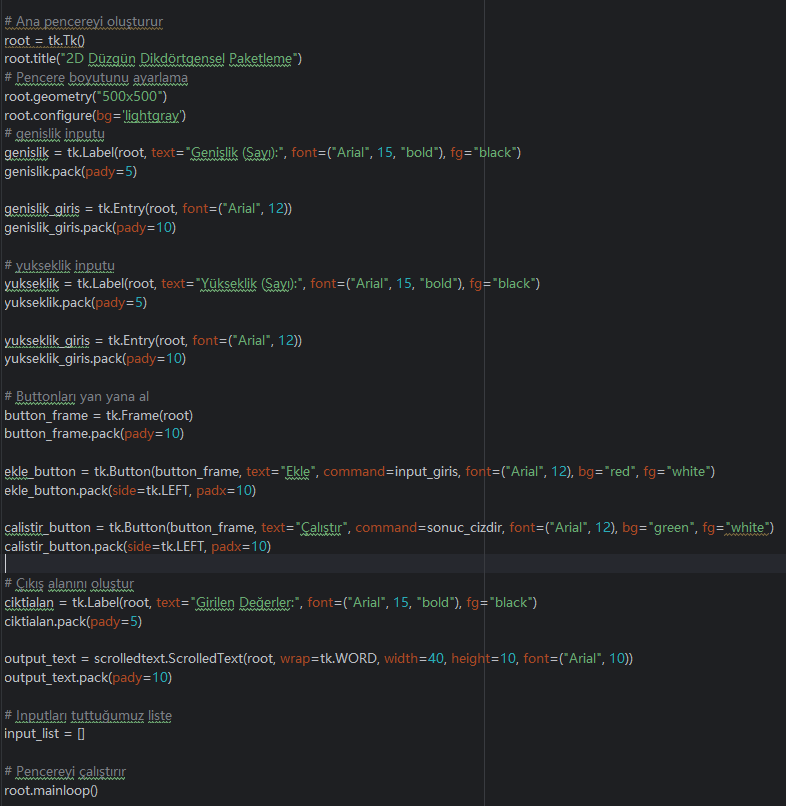
çizdir adlı fonksiyonda ise bu parçaların plaka üzerindeki yerleşiminin görselleştirilmesi matplotlitb kütüphanesi ile yapılıyor ax.set() metodu ile grafikteki değer aralıkları ayarlanıyor. ax.grid() metodu ile grafik cizgieri ekleniyor. Grafiğe eklenecek olan dikdörtgenlere rastgele renkler veriliyor.Ve pencerenin çalışması saglanıyor.

Oluşturduğumuz sonuç\_cizdir fonksiyonu ile sonuç ve cizdir fonksiyonlarını tek bir fonksiyon altında çağırıyoruz.

**metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Arayüz tasarımını tkinter kullanarak ana oluşturup boyutunu, rengini ve başlığını ayarladım .Daha sonra veri girişi için 2 tane label ekleyerek dikdörtgenin genişlik ve yükseklik değerlerini aldım. Ekle ve Çalıştır butonları ekledim. Kullanıcın girdiği değerleri görmesi için girilen değerler adlı label oluşturup orada gösterdim.

****

Program**ı** çalıştırdığımızda ilk olarak ana pencere açılıyor. Ve açılan pencerede veri setinden değerler girerek Ekle butonuna basıyoruz.(veri setini sadece test amacıyla çıktı almak için kullandım)

**metin, ekran görüntüsü, ekran, görüntüleme, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Eklenen değerler Girilen Değerler kısmına yazdırıyor.

**metin, ekran görüntüsü, ekran, görüntüleme, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu**

Diğer dikdörtgen verilerini de girdikten sonra Çalıştır butonuna basıyoruz ve karşımıza 2 pencere çıkıyor.

**metin, elektronik donanım, ekran görüntüsü, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldumetin, ekran görüntüsü, dikdörtgen, kare içeren bir resim

Açıklama otomatik olarak oluşturuldu**

İlk pencere Sonuçlar penceresinde dikdörtgen parçaların genişlik ve yükseklik değerlerini , kullanılan parçaları ve onlara ait en uygun yerleşim koordinatını gösteriyor. Diğer ikinci pencere ise bu parçaların plaka üzerindeki yerleşiminin görselini gösteriyor.(Bu örnek çıktı görselleri 2. Veri seti verileri verilerek elde edilmiştir)

**5-** Sonuç olarak bu projede verilen parçaların belirli bir plaka içine en uygun şekilde yerleştirilmesini sağlayan bir optimizasyon algoritması geliştirmeyi , tkinter kütüphanesi kullanarak kullanıcı arayüzü oluşturmayı, matplotlib kütüphanesi kullanarak bir grafik çizim yapmayı ve karşılaştığım bir çok hatayı çözmeyi öğrendiğimi söyleyebilirim.

**6- Kaynaklar:**

[**https://dspace.balikesir.edu.tr/xmlui/handle/20.500.12462/1456**](https://dspace.balikesir.edu.tr/xmlui/handle/20.500.12462/1456)

**Erişim Tarihi:20.01.2024**

[**A two-level search algorithm for 2D rectangular packing problem - ScienceDirect**](https://www.sciencedirect.com/science/article/abs/pii/S0360835207000678?via%3Dihub)

**Erişim Tarihi:20.01.2024**

[**https://www.csc.liv.ac.uk/~epa/surveyhtml.html**](https://www.csc.liv.ac.uk/~epa/surveyhtml.html)

**Erişim Tarihi:20.01.2024**

[**https://www.sciencedirect.com/science/article/pii/S1877050913003980**](https://www.sciencedirect.com/science/article/pii/S1877050913003980)

**Erişim Tarihi:10.01.2024**

<https://www.dei.unipd.it/~fisch/ricop/tesi/tesi_dottorato_Lodi_1999.pdf>

**Erişim Tarihi:16.01.2024**

[**https://visualtk.com/**](https://visualtk.com/) **Erişim Tarihi:21.01.2024**

[**https://docs.python.org/3/library/tkinter.html**](https://docs.python.org/3/library/tkinter.html) **Erişim Tarihi:21.01.2024**

[**https://matplotlib.org/**](https://matplotlib.org/) **Erişim Tarihi:19.01.2024**

[**https://www.w3schools.com/python/matplotlib\_pyplot.asp**](https://www.w3schools.com/python/matplotlib_pyplot.asp)

**Erişim Tarihi:19.01.2024**

[**https://pypi.org/project/matplotlib/**](https://pypi.org/project/matplotlib/) **Erişim Tarihi:19.01.2024**

[**https://www.lucidchart.com/pages/landing?utm\_source=google&utm\_medium=cpc&utm\_campaign=\_chart\_en\_tier3\_mixed\_search\_brand\_exact\_&km\_CPC\_CampaignId=1484560207&km\_CPC\_AdGroupID=60168114711&km\_CPC\_Keyword=lucidchart%20download&km\_CPC\_MatchType=e&km\_CPC\_ExtensionID=&km\_CPC\_Network=g&km\_CPC\_AdPosition=&km\_CPC\_Creative=442433234375&km\_CPC\_TargetID=kwd-465974767562&km\_CPC\_Country=1012766&km\_CPC\_Device=c&km\_CPC\_placement=&km\_CPC\_target=&gad\_source=1&gclid=Cj0KCQiAwbitBhDIARIsABfFYILXEsbXnRpVbntQIIh3em6wfjb6ofkImVylaVFzMzmFnM0hAA4oxBkaAgKKEALw\_wcB**](https://www.lucidchart.com/pages/landing?utm_source=google&utm_medium=cpc&utm_campaign=_chart_en_tier3_mixed_search_brand_exact_&km_CPC_CampaignId=1484560207&km_CPC_AdGroupID=60168114711&km_CPC_Keyword=lucidchart%20download&km_CPC_MatchType=e&km_CPC_ExtensionID=&km_CPC_Network=g&km_CPC_AdPosition=&km_CPC_Creative=442433234375&km_CPC_TargetID=kwd-465974767562&km_CPC_Country=1012766&km_CPC_Device=c&km_CPC_placement=&km_CPC_target=&gad_source=1&gclid=Cj0KCQiAwbitBhDIARIsABfFYILXEsbXnRpVbntQIIh3em6wfjb6ofkImVylaVFzMzmFnM0hAA4oxBkaAgKKEALw_wcB) **Erişim Tarihi:22.01.2024**