

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulmak

Github Adresi

<https://github.com/ahmetirmak06/PythonInPython>

Grubu Oluşturan Öğretmenlerin Listesi

Ahmet IRMAK, Alp Eren DEMİRKAN, Bekir PALAZ, Binnur SANCAK PALAZ, Emrah ŞEKER, Murat KOÇALI

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulmak

Analiz Raporu

Kısa Özet

Yapay Zekâ çalışmalarının içinde önemli bir yere sahip olan sezgisel algoritmalar yardımıyla, birçok gömülü sistem içinde küçük ama önemli bir parça olarak yer alan denklem kökü bulma/değişken değeri hesaplama işlevi için, yeni öğrenmekte olduğumuz Python dilini kullanmakta kendimizi geliştirmek istiyoruz. Bu şekilde Python kütüphanesinde yer alan modüllerden yararlanmayı öğrenmek ve kendimizin ürettiği alternatif bir modül de sunarak yeni bir modül kazandırmak şeklinde bir amacı takip edeceğiz.

Problem Tanımı

Python dili, yapay zekâ çalışmalarında, özellikle de sezgisel algoritmalar tarafında hangi paket/modül kütüphanelerine sahiptir? Bunlar nasıl kullanılır? Bu kütüphanelere temel işlevler açısından kullanışlı bir modül nasıl eklenir? Açık kaynak kodlu bir dil olarak Python' un diğer dillere göre geliştirme süreci/kodlama/performans açılarından nasıl bir üstünlüğü vardır?

Analiz Süreci

1. İhtiyaç Analizi

Başta gömülü sistemler olmak üzere yapay zeka tekniklerinin kullanıldığı yazılımlar günümüzde hem hızla gelişmekte hem de sistem performansları açısından önemli bir yeri vardır. Bu tür yazılımların kullanıldığı sistemlerde dışarıdan girilen/okunan bir takım veriler üzerinden bazı değişkenleri tekrar hesaplamak, "an" seviyesinde kısa zamanlar içerisinde buna uygun çıktılar üretmek sıkça kullanılan bir işlemdir. Python' un yeni ve gelişime açık bir yazılım dili olarak, bu tür temel işlevlere yüksek performanslı bir katkı sağlaması beklenir. Bu anlamda hangi kütüphanelerle çalışılabilir, geliştirdiğimiz kütüphaneleri nasıl kullanıma sunabiliriz, geçmişteki kodlamalara göre nasıl bir performans iyileştirmesi gerçekleştirebiliriz? Bu alanda çalışanlar açısından daha iyi performanslı bir algoritma geliştirmek önemli bir hedeftir. Grup olarak geliştireceğimiz bir yazılım ile konuyu farklı bir açıdan ele alacağız.

2. İçerik Analizi

Projemiz için oluşturduğumuz grubumuz, whatsapp üzerinden iletişim sağlarken, aynı zamanda video konferans araçları ile de bir araya gelecektir. Github hesabındaki paylaşım ile de projemizin hangi aşamada olduğu takip edilebilecektir.

Çalışma sonunda ortaya çıkacak ürünümüz, Python'un paylaşım platformlarında geliştirilmeye açık bir kütüphanenin başlangıç noktası olarak paylaşarak, proje bu aşama için sonlandırılacaktır.

3. Durum Ortam Analizi

Arayüz tasarımı için *PyQt5*, *Tkinter*, gerekirse veri tabanı için *Sqlite3*, temel işlemler için *sys* ve *os*, ileri düzey matematiksel işlev gerekmesi durumunda *Numpy* kullanılacaktır.

4. Kullanıcı Analizi

Bu proje kapsamında geliştireceğimiz yazılım, GitHub'a yüklenerek, özellikle yapay zeka alanında çalışan akademisyen ve yazılımcıların ulaşımına sunulacaktır. Umarız gelişen bir kütüphanenin başlangıcı veya gelişmekte olan bir kütüphanenin parçası olur ve çok sayıdaki projeye katkı sunar.

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulmak

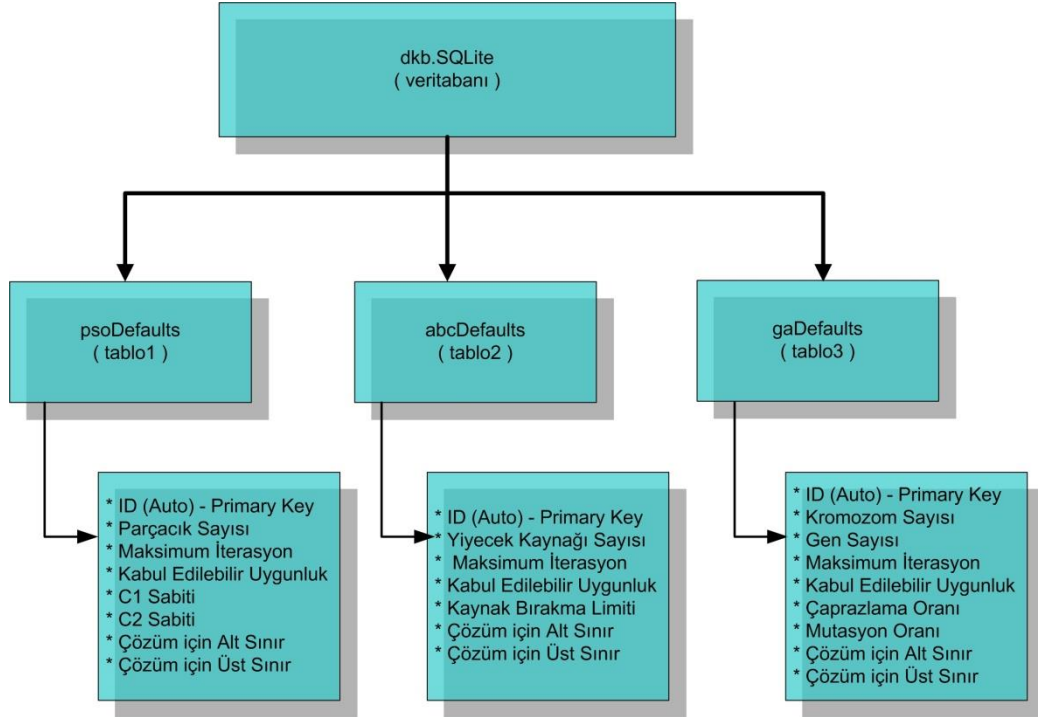
Tasarım Raporu

Kısa Özet

- ✓ Grup üyelerinin iletişimi için anlık mesajlaşma grubu kuruldu,
- ✓ Proje olarak ne yapılabilir konusunda beyin fırtınası yapıldı, projeye karar verildi,
- ✓ Projenin yapım aşamaları konusunda fikir alışverişinde bulunuldu,
- ✓ Grup üyeleri arasında görev paylaşımı yapıldı,
- ✓ Sürece göre zaman planlaması yapıldı,

| Üye Adı | Görevi |
|---------------------|--|
| Ahmet IRMAK | Sezgisel Algoritma modüllerinin yazılması |
| Alp Eren DEMİRKAN | Veritabanı okuma yazma modüllerinin yazılması |
| Bekir PALAZ | Veritabanının tasarımı ve oluşturulması. Numpy kütüphanesinin araştırılması |
| Binnur SANCAK PALAZ | Veritabanının tasarımı ve oluşturulması. Numpy kütüphanesinin araştırılması |
| Emrah ŞEKER | Denklem giriş ara yüzüne ait modülün yazılması |
| Murat KOÇALI | Genel ara yüzün tasarımı. Python' da geliştirilmiş diğer yapay zekâ alanıyla ilgili kütüphanelerinin araştırılması |

Veri Tasarımı



Ara yüz Tasarımı

Sezgisel Algoritmalar

PARTICLE SWARM OPTIMIZATION - Parçacık Sürü Optimizasyonu

Parçacık Sayısı

4

Maksimum İterasyon

100

Kabul Edilebilir Uygunluk

0,95

C1 Sabiti

1,3

C2 Sabiti

1,3

Çözüm için Alt Sınır

0

Çözüm için Üst Sınır

10

default OKU

default olarak KAYDET

En iyi çözüm X

Uygunluklar

İterasyon Sayısı

PSO ile ÇÖZ

Denklem GİR

GENETIC ALGORITHM - Genetik Algoritma

Kromozom Sayısı

30

Gen Sayısı

8

Maksimum İterasyon

500

Kabul Edilebilir Uygunluk

0,95

Çaprazlama Oranı

0,95

Mutasyon Oranı

0,1

Çözüm için Alt Sınır

0

Çözüm için Üst Sınır

10

default OKU

default olarak KAYDET

En iyi çözüm X

Uygunluklar

İterasyon Sayısı

GA ile ÇÖZ

Denklem GİR

ARTIFICIAL BEE COLONY - Yapay Arı Kolonisi

Yiyecek Kaynağı Sayısı

30

Maksimum İterasyon

100

Kabul Edilebilir Uygunluk

0,95

Kaynak Bırakma Limiti

10

Çözüm için Alt Sınır

0

Çözüm için Üst Sınır

10

default OKU

default olarak KAYDET

En iyi çözüm X

Uygunluklar

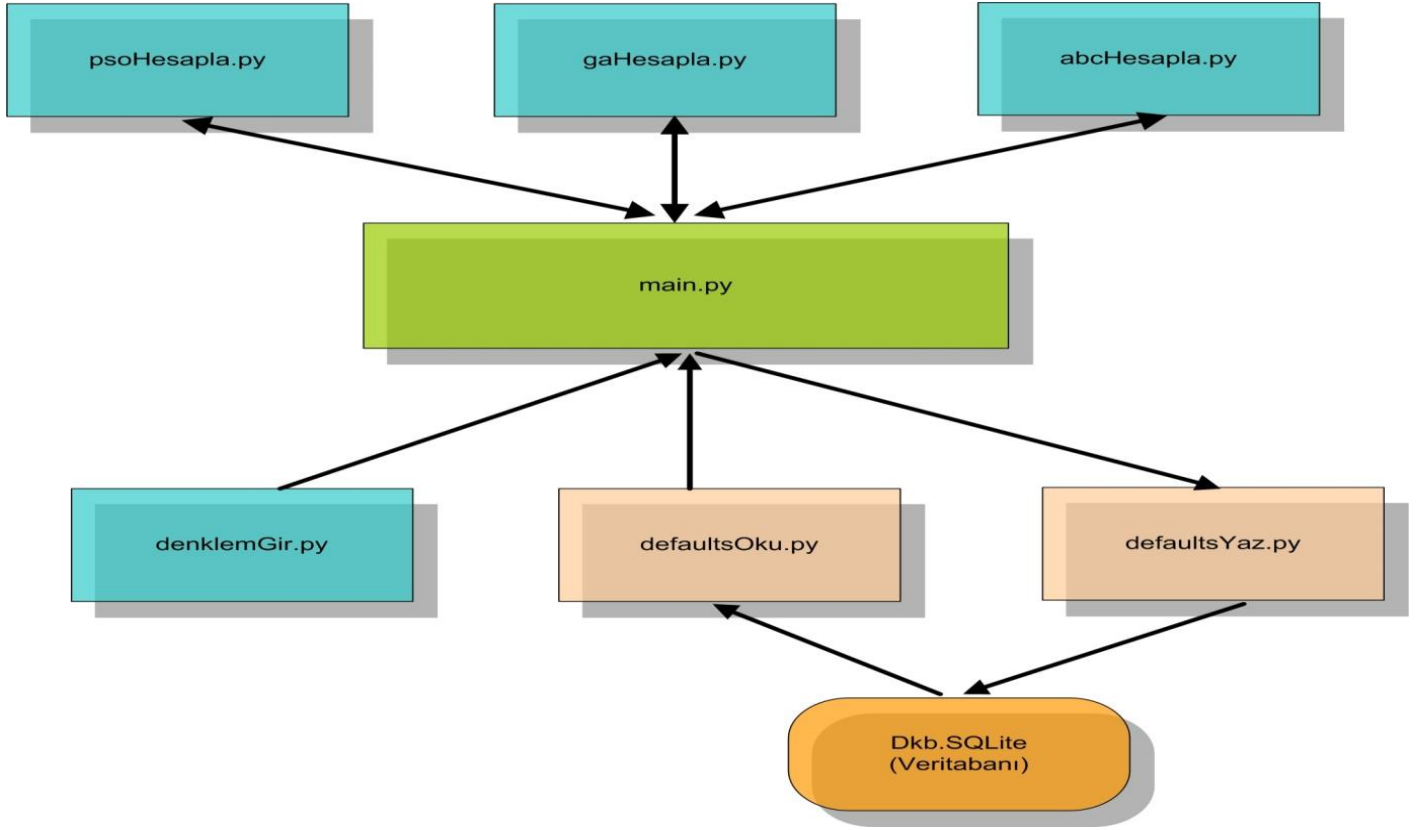
İterasyon Sayısı

ABC ile ÇÖZ

Denklem GİR

ÇIKIŞ

Kod Tasarımı



Zaman Çizelgesi

| Sıra No | Bölüm Adı | Kim Tarafından Yapılacağı | Süresi (Gün) | Başlangıç | Bitiş |
|---------|--|---------------------------|--------------|------------|------------|
| 1 | Proje konusunun araştırılması | Tüm Ekip Üyeleri | 11 | 20/11/2020 | 30/11/2020 |
| 2 | Proje konusunun tartışılması ve karar aşaması | Tüm Ekip Üyeleri | 11 | 20/11/2020 | 30/11/2020 |
| 3 | İş bölümünün yapılması | Tüm Ekip Üyeleri | 11 | 20/11/2020 | 30/11/2020 |
| 4 | Sezgisel Algoritma modüllerinin yazılması | Ahmet IRMAK | 16 | 7/12/2020 | 22/12/2020 |
| 5 | Veritabanı okuma yazma modüllerinin yazılması | Alp Eren DEMİRKAN | 13 | 6/12/2020 | 18/12/2020 |
| 6 | Veritabanının tasarımı ve oluşturulması. Numpy kütüphanesinin araştırılması | Bekir PALAZ | 14 | 2.12.2020 | 15.12.2020 |
| 7 | Veritabanının tasarımı ve oluşturulması. Numpy kütüphanesinin araştırılması | Binnur SANCAK PALAZ | 14 | 2.12.2020 | 15.12.2020 |
| 8 | Denklem giriş ara yüzüne ait modülün yazılması | Emrah ŞEKER | 21 | 2.12.2020 | 22.12.2020 |
| 9 | Genel ara yüzün tasarımı. Python' da geliştirilmiş diğer yapay zekâ alanıyla ilgili kütüphanelerinin araştırılması | Murat KOÇALI | 8 | 5.12.2020 | 12.12.2020 |
| 10 | Uygulamanın birleştirilmesi, test edilmesi ve github a yüklenmesi | Tüm Ekip Üyeleri | 5 | 23.12.2020 | 27.12.2020 |

| | EKİP ÜYELERİ | PROJE BÖLÜMLERİ | SÜRE (GÜN) | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---------------------|--|------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| | | | 20.11.2020 | 30.11.2020 | 1.12.2020 | 2.12.2020 | 3.12.2020 | 4.12.2020 | 5.12.2020 | 6.12.2020 | 7.12.2020 | 8.12.2020 | 9.12.2020 | 10.12.2020 | 11.12.2020 | 12.12.2020 | 13.12.2020 | 14.12.2020 | 15.12.2020 | 16.12.2020 | 17.12.2020 | 18.12.2020 | 19.12.2020 | 20.12.2020 | 21.12.2020 | 22.12.2020 | 23.12.2020 | 24.12.2020 | 25.12.2020 | 26.12.2020 | 27.12.2020 |
| PROJE BÖLÜMLERİ | Tüm Ekip Üyeleri | Proje konusunun araştırılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Tüm Ekip Üyeleri | Proje konusunun tartışılması ve karar aşaması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Tüm Ekip Üyeleri | İş bölümünün yapılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Ahmet IRMAK | Sezgisel Algoritma modüllerinin yazılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Alp Eren DEMİRKAN | Veritabanı okuma yazma modüllerinin yazılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Bekir PALAZ | Veritabanının tasarımı ve oluşturulması. Numpy kütüphanesinin araştırılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Binnur SANCAK PALAZ | Veritabanının tasarımı ve oluşturulması. Numpy kütüphanesinin araştırılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Emrah ŞEKER | Denklem giriş ara yüzüne ait modülün yazılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Murat KOÇALI | Genel ara yüzün tasarımı. Python’ da geliştirilmiş diğer yapay zekâ alanıyla ilgili kütüphanelerinin araştırılması | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | Tüm Ekip Üyeleri | Uygulamanın birleştirilmesi, test edilmesi ve github a yüklenmesi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

*Proje gerçekleştirme zaman çizelgesini incelemek için lütfen yaklaşınız.

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Gerçekleştirme Raporu

Karşılaşılan Sorunlar ve Uygulanan Çözümler

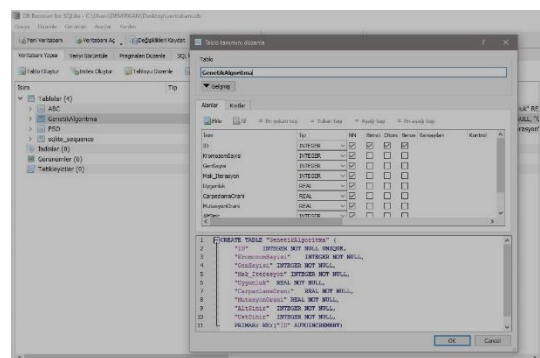
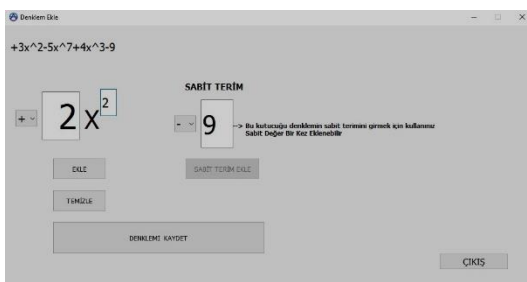
Proje aşamasında karşılaşılan pek çok problem ile ilgili internet üzerinden birçok kaynak taranarak çözüme ulaşılmaya çalışıldı. Ulaşılamayan yerlerde grup üyeleri ile eşgüdüm sağlandı.

Proje Bileşenleri ve Görevleri

Ortam Phyton 3.5, Anaconda, PyCharm, SQLite3 veritabanı kullanıldı. SQLite browser, wt designer tasarımlar için kullanıldı. Kütüphaneler; math, pyqt5, Os kullanıldı.

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulma programımızın kullanıcı arayüzü

- ✓ Çözüm Ekranı (PSO/GA/ABC)
- ✓ Denklem giriş ekranı
- ✓ Veritabanı kayıt ekleme/silme ekranı



Github Yükleme Süreci

Grup üyeleri oluşturdukları github hesapları ile 1 repostory 1 project (dkb) üzerinden yazılım kaynak kodunu birbirleri ile paylaşarak geliştirmektedir. Eğitimcilerce duyurulacak hesaplarla da paylaşım yapılacaktır.

Yazılım kaynak kodunun github profiline yükleme işlemini tek bir kullanıcı üzerinden gerçekleştirdik.

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Test Raporu

Karşılaşılan Sorunlar ve Uygulanan Çözümler

Proje aşamasında karşılaşılan problemlerle ilgili internet üzerinden birçok kaynak taranarak çözüme ulaşılmaya çalışıldı. Ulaşılamayan yerlerde grup üyeleri ile eşgüdüm sağlandı. Önceki programlama dillerinden farklı yazım kuralları sebebiyle alışma döneminde güçlük çekilmiştir (İleri ket vurma). Özellikle grafik ara yüzü ile tasarım yapılmasında, program ara yüzüne eklemek istediğimiz özellikler ile ilgili yazılan kod çalışmamış; ardından wt designer kullanılarak aynı kodlar yazılmış ve çalışmıştır. Yaşanan zorluklar bu tip hata ayıklamalar üzerinde odaklanmıştır.

Test Sürecinde Kullanılan Modüller (Varsa)

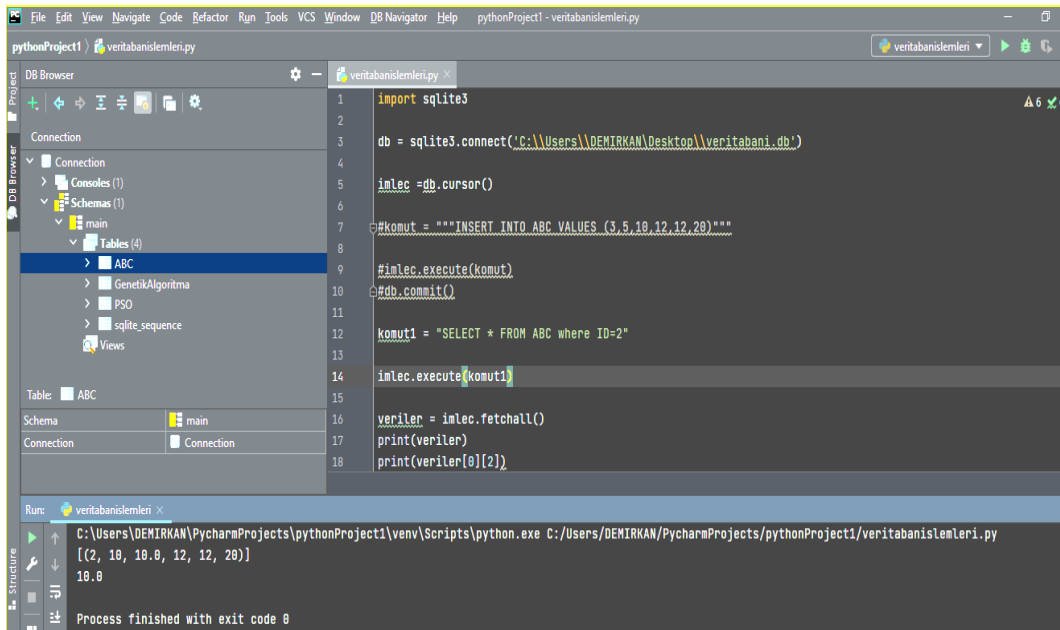
Proje test sürecinde ekstra herhangi bir modül kullanılmamıştır. Bu raporun teslimi tarihi itibarıyla eksikler tamamlanmaya çalışılmaktadır. Proje son aşamasına getirilemediğinden, test aşaması henüz yapılamamıştır. Test aşaması tamamlandığında, nihai raporda bu kısma yer verilecektir.

Ortam Phyton 3.5, Anaconda, PyCharm, SQLite3 veritabanı kullanıldı. SQLite browser, wt designer tasarımlar için kullanıldı. Kütüphaneler; math, pyqt5, os, random kullanıldı.

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulma programımızın kullanıcı ara yüzü

- ✓ Çözüm Ekranı (PSO/GA/ABC)
- ✓ Denklem giriş ekranı
- ✓ Veritabanı kayıt ekleme/silme ekranı

Yaptığımız projenin ekran görüntüleri ve kodlamalar:



```
1 import sqlite3
2
3 db = sqlite3.connect('C:\\Users\\DEMIRKAN\\Desktop\\veritabani.db')
4
5 inlec = db.cursor()
6
7 #komut = """INSERT INTO ABC VALUES (3,5,10,12,12,20)"""
8
9 #inlec.execute(komut)
10 #db.commit()
11
12 komut1 = "SELECT * FROM ABC where ID=2"
13
14 inlec.execute(komut1)
15
16 veriler = inlec.fetchall()
17 print(veriler)
18 print(veriler[0][2])
```

Run: veritabanislenleri

C:\\Users\\DEMIRKAN\\PycharmProjects\\pythonProject1\\venv\\Scripts\\python.exe C:/Users/DEMIRKAN/PycharmProjects/pythonProject1/veritabanislenleri.py

[(2, 10, 10.0, 12, 12, 20)]

10.0

Process finished with exit code 0

Sergisel Algoritmalar

PARTICULAR SWARM OPTIMIZATION - Parçacık Sürü Optimizasyonu

Parçacık Sayısı: En iyi çözüm X: Uygunluklar:

Maksimum İterasyon: İterasyon Sayısı:

Kabul Edilebilir Uygunluk:

C1 Sabiti:

C2 Sabiti:

Çözüm için Alt Sınır:

Çözüm için Üst Sınır: +2x""=2+2

Denklem Gir

GENETIC ALGORITHM - Genetik Algoritma

Kromozom Sayısı: En iyi çözüm X: Uygunluklar:

Gen Sayısı: İterasyon Sayısı:

Maksimum İterasyon:

Kabul Edilebilir Uygunluk:

Çaprazlama Oranı:

Mutasyon Oranı:

Çözüm için Alt Sınır:

Çözüm için Üst Sınır:

Denklem Gir

ARTIFICIAL BEE COLONY - Yapay Arı Kolonisi

Yiyecek Kaynağı Sayısı: En iyi çözüm X: Uygunluklar:

Maksimum İterasyon: İterasyon Sayısı:

Kabul Edilebilir Uygunluk:

kaynak Bırakma Limiti:

Çözüm için Alt Sınır:

Çözüm için Üst Sınır:

Denklem Gir

```

File Edit View Navigate Code Refactor Run Tools VCS Window Help dkb - C:\Users\Ahmet\Desktop\dkb\venv\Lib\SA\abc.py - PyCharm
mainDkb.py x pso.py x abc.py x myWidgets.py x
29
30 def __generateCandidateFoodSources(self):
31     k = []
32     for i in range(self.__ncfs):
33         xValue = random.uniform(self.__ll, self.__ul)
34         fv = self.__fitnessValue(xValue)
35         k.append([fv, xValue, 0])
36     return k
37
38 def __fitnessValue(self, p):
39     return 1 / (1 + math.fabs(self.__polynom(p)))
40
artificialBeeAlgorithm
Run: mainDkb x
C:\Users\Ahmet\Desktop\dkb\venv\Scripts\python.exe C:/Users/Ahmet/Desktop/dkb/venv/mainDkb.py
1.00 3.0 1
Process finished with exit code 0

```

```

DB Browser
Connection
Consoles (1)
Schemas (1)
main
Tables (4)
ABC
GenetikAlgoritma
PSO
sqlite_sequence
Views
Table: ABC
Schema: main
Connection: Connection
veritabanisimleri.py x denklemler.py x
4
5 class Ui_DenklemEkle(object):
6     topLam = 0
7
8     def setupUi(self, DenklemlerEkle):
9         DenklemlerEkle.setObjectName("DenklemlerEkle")
10        icon = QtGui.QIcon()
11        icon.addPixmap(QtGui.QPixmap("sa.png"), QtGui.QIcon.Normal, QtGui.QIcon.Off)
12        DenklemlerEkle.setWindowIcon(icon)
13        DenklemlerEkle.resize(1000, 500)
14        DenklemlerEkle.setMinimumSize(QtCore.QSize(1000, 500))
15        DenklemlerEkle.setMaximumSize(QtCore.QSize(1000, 500))
16        DenklemlerEkle.setBaseSize(QtCore.QSize(0, 0))
17        DenklemlerEkle.setContextMenuPolicy(QtCore.Qt.NoContextMenu)
18
19        self.centralwidget = QtWidgets.QWidget(DenklemEkle)
20        self.centralwidget.setObjectName("centralwidget")
21
22        self.label = QtWidgets.QLabel(self.centralwidget)
23        self.label.setGeometry(QtCore.QRect(150, 150, 31, 51))
24        self.label.setLayoutDirection(QtCore.Qt.LeftToRight)
25        self.label.setStyleSheet("font: 30pt 'MS Shell Dlg 2';")
26        self.label.setObjectName("label")
27
28        self.txt_ksayi = QtWidgets.QLineEdit(self.centralwidget)
29        self.txt_ksayi.setGeometry(QtCore.QRect(70, 125, 71, 91))
30        font = QtGui.QFont()
31        font.setPointSize(48)

```

Değerlendirme Kriterleri

| KRİTERLER | Ekip Üyeleri | Eğitmen 1 | Eğitmen 2 | Eğitmen 3 | Ortalama |
|--|---|-----------|-----------|-----------|----------|
| 1. Analiz Raporunun Tamamlanması | Tüm Ekip Üyeleri | | | | |
| 2. Tasarım Raporunun Tamamlanması | Tüm Ekip Üyeleri | | | | |
| 3. Gerçekleştirim Raporunun Tamamlanması | Tüm Ekip Üyeleri | | | | |
| 4. Ara yüz tasarımı | Emrah ŞEKER, Murat KOÇALI | | | | |
| 5. Sezgisel Algoritma modüllerinin yazılması | Ahmet IRMAK | | | | |
| 6. Veri Tasarımı-Sınıf Tasarımı | Alp Eren DEMİRKAN, Bekir PALAZ, Binnur SANCAK PALAZ | | | | |
| 7. Programın Çalıştırılması | Tüm Ekip Üyeleri | | | | |
| 8. Yazılım Test Çalışması | Tüm Ekip Üyeleri | | | | |
| 9. GitHub'a Yüklenmesi | Ahmet IRMAK | | | | |