KOCAELİ ÜNİVERSİTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ YAZILIM LABORATUVARI 1 PROJE 2

MULTIHTHREADINGIN PYTHON DİLİNDE UYGULANIŞI VE VERI TEMIZLEME

1st Mustafa Can Öncü
Bilgisayar Mühendisliği 2. Öğretim
200202100
200202100@kocaeli.edu.tr

I. Projenin Özeti

Projemizin amacı ”Python” dilinde ”multithreading” algoritmasının gerçekleştirilmesidir. Verilen fonksiyonları gerçekleştirerek ödev isterlerini yerine getirmemiz beklenmektedir. Bu fonksiyon isterlerinden ilki input dosyasının ”csv” yapısına veri düzenlemeye uygunluğunun kontrolüdür. Bu isterlerden ikincisi ise kullanıcıdan alınacak olan (txt formatındaki input dosyasının içersine koyarak) sıralamanın gerçekleştirilmesidir Bu isterlerden üçüncüsü ise sıralanmış verilerin company, issue, bekleme süresi, istenen thread sayısı gibi özelliklerinin istenen veri yapısı ile istenilen formata sokulmasını sağlamaktadır. Bu yapı hem iç içe döngüler hem de if blokları yardımıyla gerçekleştrdim. Sonuncu fonksiyonumuz ise bizden verilen tüm şartların sonucunu terminal ya da tkinter ile ekrana aktarmamızdır. Konsol sürekli ”command C” tuşu yardımıyla kendini yenilemektedir ve böylece proje tamamlanmaktadır. .Raporumuz LaTeXformatında yazılmıştır.

II. GİRİŞ

A. Asıl projeye ait bilgilerin detaylarına giriş kısmı

Projenin yazıldığı IDE süreç boyunca farklılık göstermektedir. ”VSCODE” ve ”PyCharm” temel olarak kullanılan IDE’ler olarak belirlendi. Bunun yanı sıra web üzerinden online editörler de kullanılmıştır. Temel amacımız algoritmayı efektif bir şekilde kurup, bu mantığı konsola aktarma aşamasında başarılı olabilmektir.

III. TEMEL BİLGİLER

Projemizde pek çok for yapısı tanımlandı ve kullanıldı. Bu sayede ”multithreading” oluşturacağımız yapı mevcut hale geldi. ”int eklenecek olanlar urunadi;” fonksiyonunu etkileyecek işlemler yapabilmek için şu fonksiyonları tanımladık:yavasThreadlerimProduct()(), ortaThreadlerimProduct(), ertele(), add(). Otomatik çalışma sayesinde yukarıda verilen fonksiyonlara erişim sağlanarak kullanıcının isteğine göre çalıştırılması sağlandı.

IV. YÖNTEMLER

Projemizde veri yapıları ve algoritmaları dersini baz olarak çeşitli yöntemler uyguladık. Fonksiyonlarımızın hepsi tam olarak

çalışmaktadır. Bu sayede istenilen yapı gerçekleşmiştir.

A. Projenin çalışma prensibini gösteren kısım

Bu bölümde fonksiyonlar tek tek ele alınacaktır. def product():product degerini alir ve diger tüm product satırlarıyla kıyaslar .def company():company degerini alir ve diger tüm company satırlarıyla kıyaslar .def issue():issue degerini alir ve diger tüm issue satırlarıyla kıyaslar .def karisikCagirProduct() threadlerimi karişık şekilde çağırır aynı şekilde diğer variablelar icin de geçerlidir.

V. DENEYSEL SONUÇLAR

Kullanıcı programı çalıştırdığında input dosyasından çekilmiş tek bir verinin konsol ekranında yazması ile karşılaşmaktadır. Multitherading mantığı işlemler tanımlanınca görülebilir. Yapılmak istenen işlemlerin altında açıklamaların olduğu menü ile karşılaşıyoruz. Bu sayede işlemlerin gerçekleşeceği menüyü görüyoruz. İstenen tuşu ile sonuçları ekrana bastırılır. Veriler tam anlamıyla bitince son ”comman c veya q” programdan çıkışı sağlar.

A. Projenin deneysel sonucunu gösteren kısım

VI. SONUÇLAR

A. Projenin sonucunu gösteren kısım

Projemiz başarılı bir şekilde sonuca ulaştı. Bizden istenen fonksiyonlar istenen yapılar kullanılarak gerçekleştirildi. Bu proje sayesinde pythonun yapısını ve web geliştirmenin yapısını çok daha detaylı inceleme ve öğrenme fırsatı bulduk. Bu algoritma yapıları dışında bir library(kütüphane) dosyası indirerek bunu kodumuza entegre etmeyi başardık. Bunun sonucu olarak öncelikli beutifulsoup yapısını for döngüsü yoluyla yapmayı kavradık ve projemizde kullanabildik.

VII. KAYNAKÇA

A. Projede yararlandığımız kaynaklar

- www.cs.bham.ac.uk/ jxb/DSA/dsa.pdf
- Çölkesen, Rıfat, Veri Yapıları Ve Algoritmaları
- https://www.youtube.com/watch?v=aPRqocoBsFQ
- https://www.geeksforgeeks.org/advanced-data-structures/
- https://www.geeksforgeeks.org/multithreading-python-set-1/
- https://www.youtube.com/watch?v=GK2Zl5DG0VM
- https://www.youtube.com/watch?v=aUKoTGnzLic
- https://www.dataquest.io/blog/multithreading-in-python/

VIII. YALANCI KOD

- BAŞLA
- gerekli libraryler import edilir
- csv dosyası import edilir
- alınan veriler isteğe göre düzenlenmek üzere işlenir
- önce istenmeyen sütunlar atılır
- sonra nltk ile ve diğer işlemlerle veri temizlenir
- temizlenen veriler için karşılaştırma döngüsü yazılır
- yazılan döngü iç içe for döngüsüne sokularak kıyaslama yapılır
- kıyaslama sonucu birinci forun sonunda print ile bastırılır
- arayüz kısmına geçilir
- tkintr import edilir
- pencere açılır
- labellar oluşturular
- butonlar oluşturulur
- thread kısmı oluşturulur
- thread kısmının içinden kıyaslama fonksiyonları çağırılır
- thread kısmı bir fonksiyona atanır
- atanan fonksiyonlar butonlara atanır
- butonlar sayesinde thread özelliği çalıştırılır
- cmd c veya q ile program sonlanabilir
- BİTİR

main.py ● rows.csv

main.py > yavasThreadlerimCompany

```
292
293 def yavasThreadlerimCompany():
294     pool = ThreadPoolExecutor(max_workers=20)
295     work1 = pool.submit(company_kiyas)
296     print("thread 1 calisiyor")
297
298 def ortaThreadlerimCompany():
299     pool = ThreadPoolExecutor(max_workers=200)
300     work1 = pool.submit(company_kiyas)
301     work2 = pool.submit(company_kiyas)
302     work3 = pool.submit(company_kiyas)
303     print("thread 2 calisiyor")
304
305 def hizliThreadlerimCompany():
306     pool = ThreadPoolExecutor(max_workers=2000)
307     work1 = pool.submit(company_kiyas)
308     work2 = pool.submit(company_kiyas)
309     work3 = pool.submit(company_kiyas)
310     work4 = pool.submit(company_kiyas)
311     print("thread 3 calisiyor")
312
313 def karisikCagirProduct():
314     print("thread1")
315     hizliThreadlerimProduct()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMİNAL JUPYTER: VARİABLES

```
1.0
105809
1.0
105788
1.0
105809
1.0
```

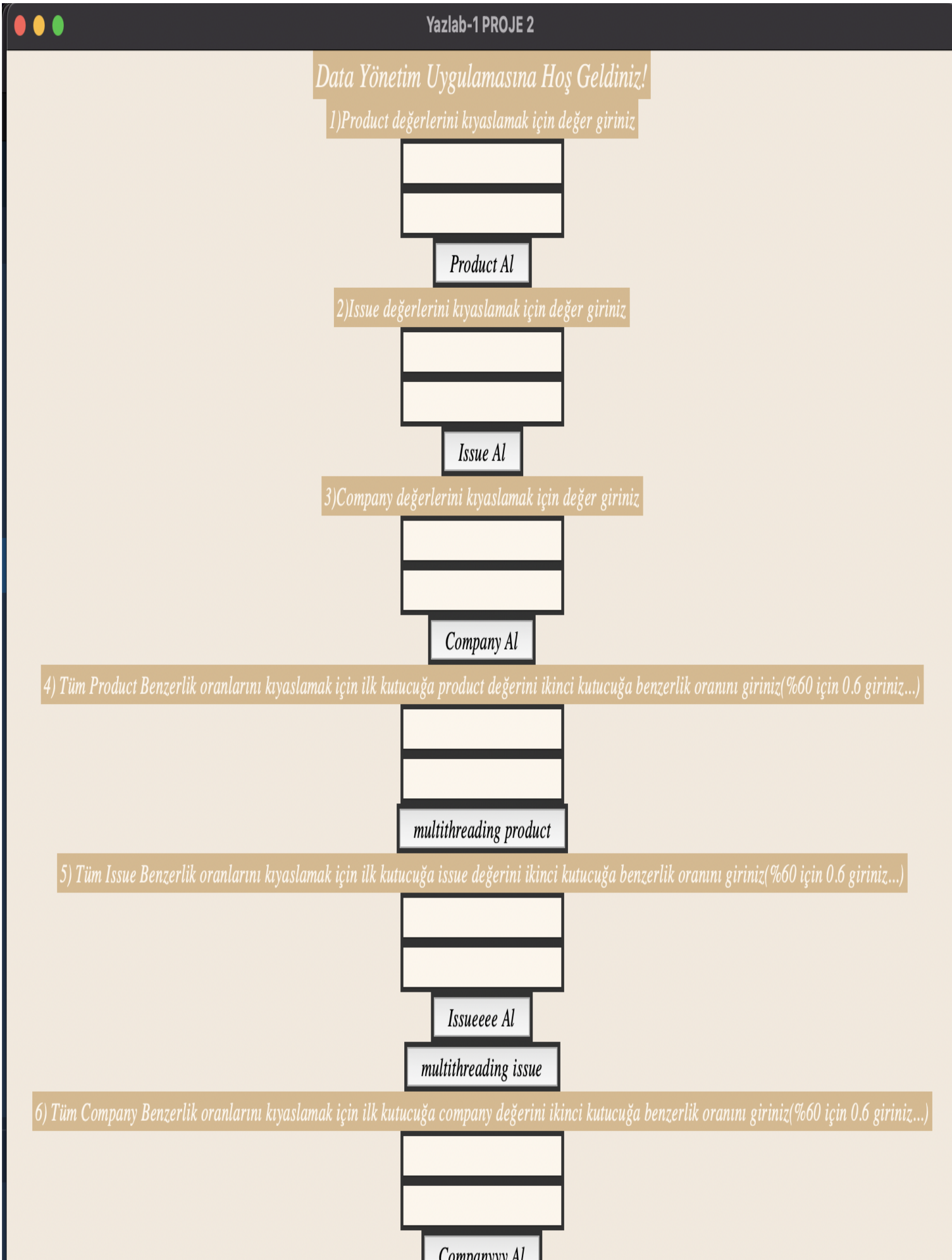
main.py X rows.csv

main.py > karisikCagirProduct

```
384 tumthreadlerbutonu.pack()
385
386 selection5 = tk.Label(text="5) Tüm Issue Benzerlik oranlarını kıyaslamak iç
387 selection5.pack()
388 entry9 = tk.Entry(text="kıyasistediğiniz ilk deeri grin",fg="black",bg="#FDF
389 entry9.pack()
390 entry10 = tk.Entry(text="kıyistediğin ikinci değerigirin",fg="black",bg="#FD
391 entry10.pack()
392 buton5 = tk.Button(form,text="Issueeee Al",fg="black",bg="#DAB88B",font="Tim
393 buton5.pack()
394 tumthreadlerbutonu2 = tk.Button(form,text="multithreading issue",fg="black",
395 tumthreadlerbutonu2.pack()
396
397 selection6 = tk.Label(text="6) Tüm Company Benzerlik oranlarını kıyaslamak i
398 selection6.pack()
399 entry11 = tk.Entry(text="kısistediğiniz ilk deeri girin",fg="black",bg="#FDF
400 entry11.pack()
401 entry12= tk.Entry(text="Oranınınızı giirlinz lutfen",fg="black",bg="#FDF6EC"
402 entry12.pack()
403 buton6 = tk.Button(form,text="Companyyy Al",fg="black",bg="#DAB88B",font="Ti
404 buton6.pack()
405 tumthreadlerbutonu3 = tk.Button(form,text="multithreading company",fg="black
406 tumthreadlerbutonu3.pack()
407
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMİNAL JUPYTER: VARİABLES

```
13                                          Debt collection
14                                          Debt collection
15                                          Debt collection
16                                Checking or savings account
17                                          Debt collection
18                                          Debt collection
19                                          Debt collection
thread 3 calisiyor
```


Yazlab-1 PROJE 2
Data Yönetim Uygulamasına Hoş Geldiniz!
1)Product değerlerini kıyaslamak için değer giriniz
Product Al
2)Issue değerlerini kıyaslamak için değer giriniz
Issue Al
3)Company değerlerini kıyaslamak için değer giriniz
Company Al
4) Tüm Product Benzerlik oranlarını kıyaslamak için ilk kutucuğa product değerini ikinci kutucuğa benzerlik oranını giriniz(%60 için 0.6 giriniz...)
multithreading product
5) Tüm Issue Benzerlik oranlarını kıyaslamak için ilk kutucuğa issue değerini ikinci kutucuğa benzerlik oranını giriniz(%60 için 0.6 giriniz...)
Issueeee Al
multithreading issue
6) Tüm Company Benzerlik oranlarını kıyaslamak için ilk kutucuğa company değerini ikinci kutucuğa benzerlik oranını giriniz(%60 için 0.6 giriniz...)
Companyyy Al