



T.C

**KOCAELİ SAęLIK VE TEKNOLOJİ ÜNİVERSİTESİ**  
**YAZILIM MÜHENDİSLİęİ PROGRAMI**

**ÖDEV KONUSU**  
**LİMAN OTOMASYON**

**Hazırlayan**

**ÖęRENCİ ADI : ŞEYMA LAÇIN**  
**ÖęRENCİ NUMARASI : 220502035**  
**ÖęRENCİ ADI: ZEHRA YARDIMCI**  
**ÖęRENCİ NUMARASI:220502038**

**Github Hesap Linkleri:**

<https://github.com/Seyma-lacin>  
<https://github.com/Zehrayardimci>

**DERS SORUMLUSU**  
**Prof.Dr. HÜSEYİN TARIK DURU**

**12.12.2023**

---

## İÇİNDEKİLER

1.	<u>ÖZET</u>	3
2.	<u>GİRİŞ</u>	3
3.	<u>YÖNTEM</u>	4
	3.1 csv Dosyalarının Okunması:.....	4
	3.2 Liman Sınıfı:.....	4
	3.3 Tır Sınıfı:.....	6
	3.4 Gemi Sınıfı :.....	7
	3.5 Stack Sınıfı:.....	7
	3.6 Liman İşlemleri Ve Simülasyon.....	8
4.	<u>SONUÇ VE ÖĞRENİLEN DERSLER</u>	9
5.	<u>KAYNAKÇA</u> .....	10

Ödev No: 2	Tarih 12.10.2023	2/10
------------	------------------	------

## 1. ÖZET

Bu çalışma, Python programlama dili kullanılarak bir liman otomasyon sistemi simülasyonunu içermektedir. Temel amacımız, limandaki yük indirme-yükleme süreçlerini simüle ederek sistemin işleyişini anlamaktır. Python dilinde yazılan program, gemi ve TIR işlemlerini başarıyla yönetmek üzere senaryo gereksinimlerini karşılamaktadır. Ancak, mevcut sistem sadece TIR'ların yük indirme işlemlerini gerçekleştirebilmekte ve gemilere yük yükleme yeteneğine sahip değildir.

## 2. GİRİŞ

Bu çalışma, Python programlama dilini kullanarak bir liman otomasyon sisteminin simülasyonunu gerçekleştirmek amacıyla tasarlanmıştır. Liman operasyonları, gemi ve TIR işlemlerini kapsayan bu simülasyon, bir dizi temel yöntem ve araç kullanılarak gerçekleştirilmektedir.

### 2.1 Temel Amacımız

Bu çalışmanın temel amacı, liman operasyonlarını simüle ederek sistemin işleyişini anlamaktır. Gemi ve TIR işlemlerinin yönetimi, limanlardaki yük indirme ve yükleme süreçlerini etkili bir şekilde modellemek üzere tasarlanmıştır.

### 2.2 Kullanılan Temel Yöntemler ve Araçlar

Bu simülasyon, aşağıdaki temel yöntemler ve araçlar kullanılarak gerçekleştirilmiştir:

- **Olay Tabanlı Simülasyon:** Liman operasyonlarını adım adım simüle etmek için olay tabanlı bir yaklaşım benimsenmiştir.
- **Python Programlama Dili:** Simülasyon programı Python dilinde yazılmıştır.
- **csv Kütüphanesi:** Gemi ve TIR bilgilerini dış dosyalardan okumak için csv kütüphanesi kullanılmıştır.

### 2.3 Beklenen işlemler

Bu kodun temel amacı, liman operasyonlarına ait temel işlemleri simüle etmektir. Ancak, mevcut kod sadece TIR'lardan yük indirme işlemlerini gerçekleştirebilmekte ve gemilere yük yükleme yeteneğine sahip değildir.

Ödev No: 2	Tarih 12.10.2023	3/10
------------	------------------	------

---

### 3. YÖNTEM

Bu bölümde kullanılan yöntemler ve bu yöntemlerin nasıl uygulandığına dair adımlar açıklanmaktadır.

#### 3.1 csv Dosyalarının Okunması

##### "olaylar.csv" Dosyasının Okunması:

- "csv" modülü içe aktarılır.
- "olaylar.csv" dosyası belirtilen dosya yoluyla açılır ve "olaylar\_file" adı altında bir dosya nesnesine atanır.
- "csv.DictReader" kullanılarak dosya satırları sözlük formatında okunur. Her satırın sütun başlıkları, sözlük anahtarları olur.
- Okunan veriler, "list()" fonksiyonu ile bir liste haline getirilir ve "olaylar\_list" adında bir değişkene atanır.

##### "gemiler.csv" Dosyasının Okunması:

- "gemiler.csv" dosyası belirtilen dosya yoluyla açılır ve "gemiler\_file" adı altında bir dosya nesnesine atanır.
- "csv.DictReader" kullanılarak dosya satırları sözlük formatında okunur. Her satırın sütun başlıkları, sözlük anahtarları olur.
- Okunan veriler, "list()" fonksiyonu ile bir liste haline getirilir ve "gemiler\_list" adında bir değişkene atanır.

#### 3.2 Liman Sınıfı

- istif\_alani\_1 ve istif\_alani\_2: İki adet istif alanını temsil eden "Stack" nesneleri. Her

Ödev No: 2	Tarih 12.10.2023	4/10
------------	------------------	------

---

biri belirli bir kapasiteye sahiptir.

- gemi\_listesi: Limana eklenen gemilerin tutulduğu liste.
- tir\_listesi: Limana eklenen TIR'ların tutulduğu liste.
- zaman: Limandaki zamanın temsil edildiği sayısal bir değer.
- aktif\_gemi: Şu anda yüklenmekte olan gemiyi temsil eden değişken.
- \_\_init\_\_(self): Liman sınıfının yapıcı metodu. Liman nesnesi oluşturulduğunda çağrılır ve limanın başlangıç durumunu ayarlar.
- limana\_tir\_ekle(self, tir): Limana bir TIR ekler. Belirli ülkelerden gelmiş TIR'ları kabul eder.
- limana\_gemi\_ekle(self, gemi): Limana bir gemi ekler. Belirli ülkelerden gelen gemileri kabul eder ve gemiyi yüklemeye başlar.
- tirlari\_yukleri\_indir(self): Limandaki TIR'ların yüklerini indirir ve istif alanına ekler.
- istif\_alani\_1\_yerlestir(self): İstif Alanı 1'deki yükleri üst üste yerleştirir.
- gemilere\_yuk\_yukle(self): Aktif gemiye, istif alanındaki yükleri yükler.
- kontrol\_et\_ve\_isle(self): Limandaki özel durumları kontrol eder ve işlemleri uygular.
- kontrol\_et\_ve\_isle\_istif\_alanlari(self): İstif alanlarındaki durumları kontrol eder ve gerekirse işlemleri uygular.
- kontrol\_et\_ve\_isle\_gemi\_yukleme(self): Gemi yükleme durumlarını kontrol eder ve gerekirse işlemleri uygular.

Ödev No: 2	Tarih 12.10.2023	5/10
------------	------------------	------

- 
- `yonlendirme_yap(self)`: İstif Alanı 1 doluysa, istif Alanı 2'ye yönlendirme yapar.
  - `istif_alani_1_is_empty(self)`: İstif Alanı 1'in boş olup olmadığını kontrol eder.
  - `istif_alani_1_is_full(self)`: İstif Alanı 1'in dolu olup olmadığını kontrol eder.
  - `limani_simule_et(self)`: Liman simülasyonunu başlatır ve TIR'ları, gemileri zaman içinde işler.
  - `liman_olustur_gemi(self, gemi_row)`: Verilen CSV satırından bir gemi nesnesi oluşturur.
  - `liman_olustur_tir(self, tir_row)`: Verilen CSV satırından bir TIR nesnesi oluşturur.

### 3.3 Tır Sınıfı

- `__init__(self, maliyet, tir_plakasi, ulke, konteyner_20, konteyner_30, yuk_miktari)` :Yapıcı Metod: TIR objesini başlatır ve TIR özelliklerini belirler.
- `maliyet`: TIR'ın maliyeti.
- `tir_plakasi`: TIR'ın plakası.
- `ulke`: TIR'ın geldiği ülke.
- `konteyner_20` ve `konteyner_30`: TIR'ın taşıdığı 20 ton ve 30 tonluk konteyner adetleri.
- `yuk_miktari`: TIR'ın taşıdığı toplam yük miktarı
- `get_yuk_bilgileri(self)` - Yük Bilgileri Al: TIR'ın taşıdığı yük bilgilerini içeren bir sözlük döndürür. İlgili bilgiler şunlardır: plaka, ülke, konteyner\_20 adeti, konteyner\_30 adeti, yük miktarı ve maliyet.

Ödev No: 2	Tarih 12.10.2023	6/10
------------	------------------	------

---

### 3.4 Gemi Sınıfı

- `__init__(self, numara, kapasite, ulke, konteyner_20, konteyner_30, yuk_miktari, maliyet)` - Yapıcı Metod: Gemi objesini başlatır ve gemi özelliklerini belirler.
- `numara`: Gemi numarası.
- `kapasite`: Geminin taşıma kapasitesi.
- `ulke`: Geminin gideceği ülke.
- `konteyner_20` ve `konteyner_30`: Geminin taşıdığı 20 ton ve 30 tonluk konteyner adetleri.
- `yuk_miktari`: Geminin taşıdığı toplam yük miktarı.
- `maliyet`: Geminin maliyeti.
- `get_yuk_bilgileri(self)` - Yük Bilgileri Al: Geminin taşıdığı yük bilgilerini içeren bir sözlük döndürür. İlgili bilgiler şunlardır: `numara`, `kapasite`, `ülke`, `konteyner_20` adeti, `konteyner_30` adeti, `yük miktarı` ve `maliyet`.
- `yuk_kapasite_dolu(self)` - Yük Kapasite Dolu mu: Geminin yük kapasitesinin dolup dolmadığını kontrol eder. Eğer yük miktarı, geminin taşıma kapasitesine eşit veya büyükse **True** döndürür, aksi takdirde **False** döndürür.

### 3.5 Stack Sınıfı

- `__init__(self, capacity)` - Yapıcı Metod: Yığındaki objeyi başlatır ve yığının kapasitesini belirler.
- `capacity`: Yığının maksimum kapasitesi.

Ödev No: 2	Tarih 12.10.2023	7/10
------------	------------------	------

- 
- `is_empty(self)` - Boş mu: Yığının boş olup olmadığını kontrol eder. Boşsa **True**, doluyrsa **False** döndürür.
  - `is_full(self)` - Dolu mu: Yığının dolu olup olmadığını kontrol eder. Doluysa **True**, boşsa **False** döndürür.
  - `push(self, item)` - Eleman Ekle: Yığına yeni bir eleman ekler.
  - `item`: Eklenen eleman.
  - `pop(self)` - Eleman Çıkar: Yığından en üstteki elemanı çıkarır ve döndürür.
  - `peek(self)` - En Üstteki Elemanı Görüntüle: Yığının en üstteki elemanını döndürür, ancak çıkarmaz.
  - `size(self)` - Eleman Sayısı: Yığındaki eleman sayısını döndürür.
  - `is_full_message(self)` - Dolu Mesajı: Yığın doluyrsa ekrana bir mesaj yazdırır.
  - `is_empty_message(self)` - Boş Mesajı: Yığın boşsa ekrana bir mesaj yazdırır.

### 3.6 Liman İşlemleri Ve Simülasyon

- `__init__(self)`: Liman sınıfının yapıcı metodu. Liman nesnesini başlatır.
- `limani_simule_et(self)`: Limanı belirli bir zaman diliminde simüle eder.
- `liman_olustur_gemi(self, olay)`: Gemi verilerini kullanarak bir Gemi nesnesi oluşturur.
- `limana_gemi_ekle(self, gemi)`: Limana bir Gemi nesnesi ekler.

Ödev No: 2	Tarih 12.10.2023	8/10
------------	------------------	------



- 
- liman\_olustur\_tir(self, olay): TIR verilerini kullanarak bir TIR nesnesi oluřturur.
  - limana\_tir\_ekle(self, tir): Limana bir TIR nesnesi ekler.
  - liman\_islemleri(olay, liman): Liman iřlemlerini gerekleřtiren fonksiyon.

#### 4. SONU VE RENİLEN DERSLER

Bu liman otomasyonu projesi, bir limandaki yk indirme-ykleme srelerini Python programlama dili kullanarak simle etmeye ynelikti. Ancak, projenin sonularına dair bazı nemli zorluklarla karřılařıldı.

Proje sresince, TIR'ların yklerini bařarıyla indirme iřlemi bařarılı bir řekilde gerekleřtirildi. Ancak, projenin nemli bir noktası olan ykleri uygun gemilere ykleyebilmek konusunda sorunlar yařandı. Yapılan testler ve analizler, gemilere yklenen konteynerlerin beklentilerle uyumlu olmadıėını gsterdi.

En byk zorluėumuz, gemilere yklenen konteynerlerin doėru bir řekilde sıralanamaması ve uygun gemilere atanamamasıydı. Bu sorunun stesinden gelmek iin eřitli stratejiler geliřtirdik. Konteynerlerin gemilere atanması srecini dzenleyen algoritmalarda ve sınıflarda eřitli deėiřiklikler yaptık. Ancak, bu abalarımıza raėmen sorun tam anlamıyla zlemedi.

Bu zorluėa raėmen, projenin geliřtirilme srecinde nemli deneyimler kazandık. Python dilinde simlasyon projelerinde alıřmanın zorluklarıyla yzleřmek, hata tespiti ve zme yeteneklerimizi geliřtirmemize olanak saėladı. Ayrıca, gemi ve TIR sınıfları arasındaki etkileřimi dzenlemek iin daha etkili bir strateji oluřturabilme becerisi kazandık.

dev No: 2	Tarih 12.10.2023	9/10
------------	------------------	------

---

## 5. KAYNAKÇA

<https://github.com/Furkan-Erdi/Python-Egitimi/blob/main/Python%20Eğitimi.ipynb>

<https://discord.com/channels/1183385795558649916/1183385795558649920/1184218737168101476>

<https://www.python.org/>

<https://www.geeksforgeeks.org/python-programming-language/>

<https://medium.com/@tolgahan.cepel/doğrusal-veri-yapıları-3-yığıt-stack-6c5db18ee934>

Ödev No: 2	Tarih 12.10.2023	10/10
------------	------------------	-------