

Programlama Laboratuvarı 1

Ömer Aran (190202012)
Umut Sütçü (200202038)

Proje 3 : Havalimanı Uçuş Yönetim Sistemi

17.12.2021

Özet

Bu doküman programlama laboratuvarı 1 dersinin 3.projesi olan havalimanı kontrol projesi için çözümü açıklamaya yönelik oluşturulmuştur.Dokümanda proje tanıtımı, çözüme yönelik yapılan araştırmalar, grafikler, kullanılan yöntemler, geliştirme ortamı ve kullanılan kaynaklara yer verilmiştir.

Projenin Tanımı

1 iniş 1 kalkış olmak üzere 2 pisti bulunan İstanbul Havalimanı'nda gün içerisinde (1-24 saat dilimi boyunca) yapılan uçuşların yönetimi için bir sistem geliştirilecektir. Havalimanında aynı anda sadece 1 uçak kalkış yapabiliyorken sadece 1 uçak iniş yapabilmektedir. Uçakların her biri iniş ve kalkışta farklı önceliklere sahiptir ve 1 günde maksimum 24 uçak iniş için izin isteyebilmektedir. Havalimanındaki uçakların öncelik sırası, iniş saati, gecikme süresi ve kalkış saati bilgileri kullanılarak; iniş pistini ve kalkış pistini kullanım sırasının belirlenmesi hedeflenmektedir.

Gerekli fonksiyonlar ile main kısmındaki kalabalık kodu kaldırıp clean code yazmaya çalıştık. Dosya okuma , satır okuma ve kuyruğa ekleme gibi fonksiyonlar ile input dosyasını okuyup, açmayı daha sonrasında ise inputtaki verileri satır satır okumak amacıyla satır okuma fonksiyonu yazdık kuyruğa belirli önceliğe göre eklendikten sonra kalkış yapan uçaklar, saatleri, erteleme süreleri ve gerekli tüm bilgileri output.txt adlı belgeye satır satır yazdırdık.

Araştırma ve Yöntemler

İlk önceliğimiz kuyruk yapısını anlamayı ve uygulamayı kavramaya çalışmak oldu, yeterli bilgiye sahip olmak için Türkçe ve İngilizce kaynaklardan yararlandık. En çok da görsel kaynaklar öncelikli kuyruğu kavramamıza yardımcı oldu. Daha sonra geliştirme ortamımızı C compiler ve IDE olan visual studio code ve code blocks olarak belirledik. Aşağıda ise öncelikli kuyruğa daha somut olması amacıyla birkaç şekil ekledik.

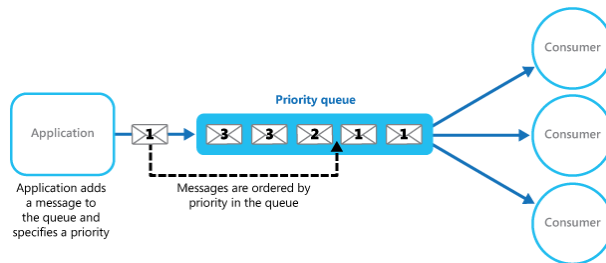


Figure 1:

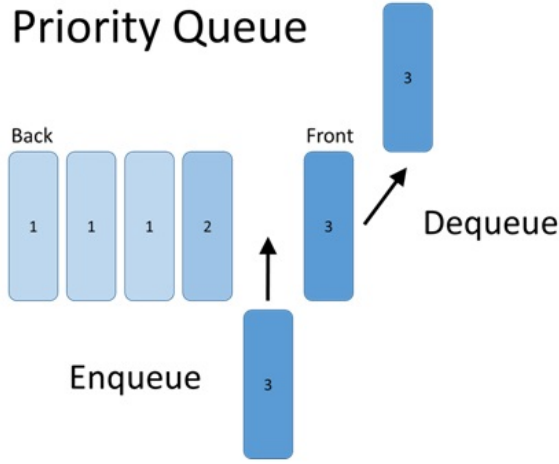


Figure 2:

Kod Bilgisi(Yalancı Kod)

Bu projede yalancı kod kullanmayı akış şemasına göre daha kullanışlı bulduğumuz için yalancı kod ile kodu daha anlaşılır şekilde hem aşağıya hem de ek'e ekledik.

```

define maxsize tanımla
global int dizi
global int kuyruk elamanları

struct ucak listesi
int 4 deger tanımlaması

global sayac

txt dosyasını okumak için fonksiyon
dosyayı aç
dosya açıldı mı kontrol et
açıldıysa
int degerleri satır satır oku
dosyayı kapat

satır satır kayıt
dosyadan okunan degerleri struct a ata

atama
for i:maxsize i artsın
satır satır dosyadan okunan degerleri ata

inis saati
int deger tanımlama
for i:maxsize
for j:maxsize-1
ucakların inis saatini karşılaştır
küçük olanı bir öncekine ata
büyük olanı bir sonrakine ata

oncelik id
int deger tanımlama
for i:maxsize
for j:maxsize-1

```

ucakların öncelik idsini karşılaştır
küçük olanı bir öncekine ata
büyük olanı bir sonrakine ata

add fonksiyonu
if kuyrukun sonu boyuttan büyükse
talep sayisini aştınız
kuyrukta yer yok

if kuyrukun başı ve sonu -1 ise
kuyruk olustur
baş ve sonu 1 arttır
satir okuma fonksiyonunu çalıştır
okunan değerleri ekrana yazdır

else
satir okuma fonksiyonunu çalıştır
okunan değerleri ekrana yazdır
sonu 1 arttır

diz fonksiyonu
inis saati fonksiyonunu çalıştır
oncelik id fonksiyonunu çalıştır

kalkış fonksiyonu
for i:jmaxsize
if ertelenme sayisi<4 ise
if talep edilen inis saati<24 ise
yeni güne geçilmiştir bastır
else
ucağın inişini onayla
for j:jmaxsize
if inen uçakla aynı saatte inecek başka uçak var
mı
aynı satte olan ucağın inis saatin ertele
erteleme sayisini 1 arttır
diz fonksiyonunu çalıştır

else
ucağın erteleme sayisi dolduğu için
farklı bir havaalanına yonlendir

output yazdirma fonksiyonu
output txt dosyası aç

if output == NULL
dosya açılmadı bastır
else
print isterleri yaz
for i:jmaxsize
ucakların bilgilerini yazdır

txt dosyasını kapat

main fonksiyonu
dosya okuma fonksiyonunu çalıştır

```
for i in range(1, n):
    add fonksiyonunu çalıştır

    diz fonksiyonunu çalıştır

    kalkis fonksiyonunu çalıştırın
output yazdırma fonksiyonunu çalıştır
```

Sonuç

Sonuç olarak kuyruk çeşitleri ve kullanım alanlarını bulup öğrenip bir programlama dilinde yazmaya ve ortaya bir havalimanı uçuş sistemi adlı projeyi çıkarmaya çalıştık. bilgisayar bilimlerinde mutlaka üzerinde durulup kavranması gereken bir proje olduğu sonucuna vardık.

Priority Queue kullanımı bir bölge, alan ya da proje yönetiminde kolaylıklar sağladığı ve ayrıca zaman anlamında da maliyeti azaltır.

Kaynakça

<https://www.youtube.com/watch?v=30yIDup0568&t=1176s>
<https://www.programiz.com/dsa/priority-queue#:~:text=A%20priority%20queue%20is%20a,their%20order%20in%20the%20queue.>
<https://www.geeksforgeeks.org/priority-queue-using-linked-list/>
<https://ubenzer.com/2009/01/cde-oncelik-kuyrugu/>

LaTeX için: <https://www.overleaf.com/learn>