

Havalimanı Simülasyonu

Ömer Emircan Ayvaz

200202009

Engin Tosun

200202028

I. ÖZET

Bu rapor Programlama Laboratuvarı I Dersinin 3. Projesini açıklamak ve sunumunu gerçekleştirmek amacıyla oluşturulmuştur. Bu proje C dilinde Code:Blocks ortamında geliştirilmiştir. Raporda projenin tanımı, özet, yöntem, karşılaşılan sorunlar ve çözümler, sözde kod, sonuç ve deneysel yöntemler bölümlerinden oluşmaktadır. Proje aşamasında yararlanılan kaynaklar raporun son bölümünde bulunmaktadır.

II. PROJE TANIMI

Bu projenin amacı Öncelikli kuyruk (priority queue) kullanarak bir havalimanı uçuş yönetim sistemi oluşturmak.

Kuyruk, ilk giren eleman ilk çıkar (First In First Out – FIFO) mantığında çalışan bir veri yapısıdır. Örneğin, kuyruk veri yapısı bankada işlem yaptırmak için sıraya girmiş insanlara benzetilebilir. Sıraya ilk giren kişi ilk işlem yaptıracaktır. Kuyruk tasarımı için dizi ya da bağlı liste kullanılabilir. Dizi kullanılan sabit boyutlu, bağlı liste kullanılarak değişken boyutlu kuyruk oluşturulabilir. Kuyrukta işlemler iki uçtan yapılır.

Öncelikli kuyruk; bazı problemlerin çözümünde doğrudan kuyruk oluşturulamaz. Örneğin; bir hastanede muayene sırasına girmiş insanlar arasında durumu acil olan birisi bulunabilir ve bu kişi muayene için öncelikli hale gelebilir. Bu gibi durumlarda öncelikli kuyruk kullanılır. Öncelikli kuyrukta ilk giren ilk çıkar mantığı geçerli değildir, önemli olan önceliklidir.

Projenin konusu: 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.

• Havalimanına iniş yapacak uçaklar öncelikle kuleden iniş yapabilmek için izin talep etmelidir.

• İniş izni talep eden her bir uçak için havalimanında

yeterli kapasite olup olmadığı kontrol edilmelidir. İniş pisti kullanım sırası öncelikli kuyruğunda yeni uçak eklemek için boş alan var mı?

• Kuleden iniş izni talep eden uçaklar için öncelikle, iniş talep edilen saatte pistin dolu mu boş mu olduğu kontrol edilmelidir. Pist boş ise iniş yapılmak istenen saate izin verilmeli ve iniş pisti kullanım sırası'nda uygun yere eklenmelidir. Aksi halde uçakların iniş sıralaması önceliğe göre belirlenmelidir.

• İniş izni talep eden her uçak için "İniş izin talebiniz onaylanmıştır veya ... nedeniyle iniş izni verilememektedir. şeklinde ekranda yazdırılmalıdır.

• Uçakların iniş ve kalkış saatleri önceliğe göre belirlenecektir. Uçakların öncelik (oncelikid) sıralaması şu şekildedir (yüksekten düşüğe):

1. Ambulans uçağı
2. Savaş uçağı
3. Yolcu uçağı
4. Kargo uçağı.

• Havalimanına iniş talep eden uçakların önceliği (oncelikid), uçak numarası (ucakid) ve talep ettiği iniş saati (talepedileninissaati) input.txt dosyasından okunacaktır.

• Tüm uçakların iniş ve kalkış süreleri eşittir ve hesaplamalara dâhil edilmeyecektir.

• Havalimanına iniş yapan her uçağın, kalkış için bekleme süresi 1 saattir. Uçakların kalkış saatine, ötelenmeden dolayı oluşan gecikme süreleri dâhil edilmelidir. Kalkış saati bu bilgiler göz önünde bulundurularak hesaplanmalıdır.

4. Sonek ağacı oluşturulan bir s katarı içinde en çok tekrar eden altkatar nedir, kaç kez tekrar etmektedir?

• Aynı önceliğe sahip iki uçak, aynı saatte kalkış yapacaksa öncelik ilk iniş yapan uçağa verilmelidir.

• Önceliği yüksek olan uçaklar nedeniyle önceliği düşük olan herhangi bir uçağın uçuşu, maksimum 3 kez ertelenebilir. Eğer 3'ten fazla ertelenme durumu söz konusuysa, öncelik gözetilmeksizin beklemede olan uçağın kalkışı gerçekleştirilmelidir.

• Kuleden bir günde maksimum 24 uçak iniş için izin talep edebilir. Eğer bu kapasite dolmuşsa;

• İniş için onay alan uçaklardan en az birinin önceliği (X uçağı olsun), iniş izni onayı bekleyen uçağın (Y uçağı olsun) önceliğinden düşüğe; yüksek öncelikli yeni uçağa (Y) iniş

onayı verilir. Daha önce onay almış ve önceliği düşük olan uçak (X) başka bir havalimanına yönlendirilmelidir.

- İniş izni daha önceden onaylanan uçağın (X) izni iptal edilmişse; “Acil iniş yapması gereken ... (Y) uçağı nedeniyle iniş izniniz iptal edilmiştir, iniş için Sabiha Gökçen Havalimanı’na yönlendiriliyorsunuz.” şeklinde ekranda yazdırılmalıdır.

- İniş izni talep eden uçakların her biri satır satır input.txt’den okunmalıdır. Okunan her bir satır ekranda gösterilmelidir.

- Her yeni input satırı okunduğunda, kalkış yapacak olan uçakların bulunduğu output.txt dosyası güncellenmeli ve güncel kalkış pisti kullanım sırası öncelikli kuyruğu ekranda gösterilmelidir.

- Projede kullanılacak input.txt dosyası ekte verilmiştir. output.txt dosyası ise proje içerisinde öğrenci tarafından oluşturulacaktır.

- input.txt dosyası aşağıdaki bilgileri içermektedir:

Havalimanına 1 günde iniş izni talep eden uçakların listesi

Dosyadan sırasıyla (satır satır) boşluk ile ayrılmış 3 bilgi (öncelik id, uçak id, talep edilen iniş saati) okunacaktır.

- output.txt dosyası uçakların kalkış saati bilgisine göre oluşturulmalıdır. Dosyanın içeriği şu şekilde olmalıdır:

Her satırda öncelik id, uçak id, talep edilen iniş saati, iniş saati, gecikme süresi, kalkış saati olmak üzere toplamda 6 bilgi içermelidir.

iniş saati bilgisi; eğer uçak talep ettiği saatte iniş yapmışsa talep edilen iniş saati ile aynı olmalıdır. Ancak talep edilen saatte önceliği yüksek başka bir uçak iniş yapacaksa belirlenen yeni iniş saati, kalkış saatinin hesaplanmasında kullanılmalıdır.

Bilgiler input.txt’de olduğu gibi boşluk ile ayrılmalıdır.

- Projede kullanılacak veri yapıları aşağıdaki gibidir:

iniş pisti kullanım sırası: Havalimanına iniş yapacak uçakların iniş pistini kullanım sırasını içeren öncelikli kuyruktur. Uçak öncelik ve iniş saati bilgisi dikkate alınarak hazırlanacaktır.

kalkış pisti kullanım sırası: Havalimanından kalkış yapacak uçakların kalkış pistini kullanım sırasını içeren öncelikli kuyruktur. Uçak öncelik ve kalkış saati bilgisine göre hazırlanacaktır.

III. YÖNTEM

Öncelikle projede kullanılacak olan öncelikli kuyruk, yığın, FOF gibi kavramlar araştırıldı. Değerler fscanf ile alınıp ondan sonra her bir değer starttan başlayarak eklendi. Eklemelemlerden sonra iniş pisti fonksiyonuna girip sıralama işlemleri yapıldı. Bu fonksiyonda önce talep edilen saate göre ondan sonra da talep edilen saatleri aynı olanlar önceliğe göre sıralandı. Bu işlemten sonra iniş izinleri verilmeye başlandı. Bittikten sonra hepsi kalkış fonksiyonuna gönderildi. Bu fonksiyonda ise istenen output çıktısına sonuçlar yazdırıldı. Aynı zamanda çıktı kontrolü için aynı değerler konsola da yazdırıldı.

IV. KARŞILAŞILAN SORUNLAR VE ÇÖZÜMLERİ

Başka Havalimanına Gönderilen Uçakların Koşul Problemi:

Hataya sebep olan durum başka havalimanına gönderilen uçağın kuyrukta kalmış gibi diğer çıktıları etkilemesi ve gecikme sürelerini değiştirmesiydi if komutuna 2 koşul daha eklenerek bu problem ortadan kaldırıldı.

V. SÖZDE KOD

```
typedef struct node
{
    int ucaak_id;
    int oncelik_id;
    int talep_edilen_inis_saati;
    int gecikme;

    struct node* next;
} ucaak;

struct node *start=NULL;

int main

    fonksiyon(kuyruk oluřtur);

void kuyruk_olustur()
    struct node *new_node,*temp;
    dosya_aç(input.txt);
    int a,b,c;
    while(fscanf(fp,"%d %d %d",&a,&b,&c)!=EOF)
        a b c yi new_node a ata;
        if(düğüm=null)
            starta a b c yi ata;
        else
            yeni düğüm oluřtur ona ata;

        new_node için malloc bellek ata;
    end while

    çağır inis_pisti_kullanım_sirasi();
    yazdır("\n\n");
    çağır guncel_kalkis_pisti_kullanım_sirasi();

    dosya_kapa(input.txt);

void inis_pisti_kullanım_sirasi()

    yazdır(inis_pisti_kullanım_sirasi);
    struct node *i,*j;
    int temp;
    for(i=start; i->next!=NULL; i=i->next)
        for(j=i->next; j!=NULL; j=j->next)
```

```

        if((i->talep_edilen_inis_saati) >
            (j->talep_edilen_inis_saati))
            yer degistir(i ve j);
    end for

    for(i=start; i->next!=NULL; i=i->next)
        for(j=i->next; j!=NULL; j=j->next)
            if((i->talep_edilen_inis_saati) >
                (j->talep_edilen_inis_saati) &&
                (i->oncelik_id) > (j->oncelik_id) )
                yer degistir(i ve j);
        end for

    for(i=start; i->next!=NULL; i=i->next)
        for(j=i->next; j!=NULL; j=j->next)
            if((i->talep_edilen_inis_saati) >
                (j->talep_edilen_inis_saati) && (i->oncelik_id)
                > (j->oncelik_id) )
                yer degistir(i ve j);
        end for

    i=start;
    while(i!=NULL)

        if(i->gecikme >3)
            yazdır(i->ucak_id başka havalimanına
                yönlendirildi);
        else
            yazdır(i->oncelik_id i->ucak_id
                i->talep_edilen_inis_saati
                i->gecikme i->talep_edilen_inis_saati+ i->gecikme);
        end while;

void guncel_kalkis_pisti_kullanım_sirasi()

yazdır(kalkış pisti kullanım sırası);
dosya_aç(output.txt);

struct node *i=start;

while(i!=NULL)

    if(i->gecikme <4)
        yazdır(i->oncelik_id, i->ucak_id,
            i->talep_edilen_inis_saati,
            i->talep_edilen_inis_saati+i->gecikme,
            i->gecikme, i->talep_edilen_inis_saati+
            i->gecikme+1);
        dosyaya_yazdır(i->oncelik_id, i->ucak_id,
            i->talep_edilen_inis_saati ,
            i->talep_edilen_inis_saati+ i->gecikme,
            i->gecikme, i->talep_edilen_inis_saati+
            i->gecikme+1);
    end while;

dosya_kapat(output.txt);

```

```

void yazdır()
    yazdır(Inis pisti kullanım sirasi\);
    struct node *i=start;
    while(i!=NULL)
        if(i->gecikme > 3)
            yazdır(ucak_id seferi gecikme kez
                ertelendiginden baska havalimanina
                yonlendirildi.);
        else
            printf(oncelik id ucak id talep edilen
                saat gecikme inis saat:);
            i=i->next;
        end while
    }

```

VI. SONUÇ

Kuyruk yapısı sadece havalimanı değil benzer birçok yapı olan banka kuyruğu , gişe kuyruğu gibi alanlarda kullanılabilir. Bazı problemlerin çözümünde doğrudan kuyruk oluşturulamaz. Örneğin; bir hastanede muayene sırasına girmiş insanlar arasında durumu acil olan birisi bulunabilir ve bu kişi muayene için öncelikli hale gelebilir. Bu gibi durumlarda öncelikli kuyruk kullanılır. Öncelikli kuyruқта ilk giren ilk çıkar mantığı geçerli değildir, önemli olan önceliklidir. Öncelikli kuyruk da kuyruk yapısının gerçek hayat problemlerindeki isteklerinden oluşmuştur. Bu yapı sıralama durumlarını çok kolaylaştırabilir.

VII. DENEYSEL YÖNTEMLER

```

Inis pisti kullanım sirasi
1 11 1 0 inis saat:1
1 12 1 1 inis saat:2
2 23 2 1 inis saat:3
2 27 2 2 inis saat:4
4 22 1 seferi 4 kez ertelendiginden baska havalimanina yonlendirildi.
3 17 1 seferi 4 kez ertelendiginden baska havalimanina yonlendirildi.
3 7 2 3 inis saat:5
4 28 3 3 inis saat:6
3 24 8 0 inis saat:8
2 13 9 0 inis saat:9
2 6 10 0 inis saat:10
2 3 11 0 inis saat:11
3 4 10 2 inis saat:12
2 2 13 0 inis saat:13
1 1 14 0 inis saat:14
3 8 14 1 inis saat:15
1 25 16 0 inis saat:16
4 10 13 seferi 4 kez ertelendiginden baska havalimanina yonlendirildi.
3 15 13 seferi 4 kez ertelendiginden baska havalimanina yonlendirildi.
3 14 16 1 inis saat:17
4 5 16 2 inis saat:18
3 21 19 0 inis saat:19
2 16 20 0 inis saat:20
4 9 17 seferi 4 kez ertelendiginden baska havalimanina yonlendirildi.
4 20 21 0 inis saat:21
4 26 20 2 inis saat:22
1 19 23 0 inis saat:23
3 18 24 0 inis saat:24

```

Kalkis pisti kullanim sirasi

```
1 11 1 1 0 2
1 12 1 2 1 3
2 23 2 3 1 4
2 27 2 4 2 5
3 7 2 5 3 6
4 28 3 6 3 7
3 24 8 8 0 9
2 13 9 9 0 10
2 6 10 10 0 11
2 3 11 11 0 12
3 4 10 12 2 13
2 2 13 13 0 14
1 1 14 14 0 15
3 8 14 15 1 16
1 25 16 16 0 17
3 14 16 17 1 18
4 5 16 18 2 19
3 21 19 19 0 20
2 16 20 20 0 21
4 20 21 21 0 22
4 26 20 22 2 23
1 19 23 23 0 0
3 18 24 24 0 1
```

```
1 11 1 1 0 2
1 12 1 2 1 3
2 23 2 3 1 4
2 27 2 4 2 5
3 7 2 5 3 6
4 28 3 6 3 7
3 24 8 8 0 9
2 13 9 9 0 10
2 6 10 10 0 11
2 3 11 11 0 12
3 4 10 12 2 13
2 2 13 13 0 14
1 1 14 14 0 15
3 8 14 15 1 16
1 25 16 16 0 17
3 14 16 17 1 18
4 5 16 18 2 19
3 21 19 19 0 20
2 16 20 20 0 21
4 20 21 21 0 22
4 26 20 22 2 23
1 19 23 23 0 0
3 18 24 24 0 1
```

Ucak id:2 talep edilen saat:13
Kalkis pisti kullanim sirasi

```
2 2 13 13 0 14
1 1 14 14 0 15
```

Ucak id:3 talep edilen saat:11
Kalkis pisti kullanim sirasi

```
2 3 11 11 0 12
2 2 13 13 0 14
1 1 14 14 0 15
```

Ucak id:4 talep edilen saat:10
Kalkis pisti kullanim sirasi

```
3 4 10 10 0 11
2 3 11 11 0 12
2 2 13 13 0 14
1 1 14 14 0 15
```

Ucak id:5 talep edilen saat:16
Kalkis pisti kullanim sirasi

```
3 4 10 10 0 11
2 3 11 11 0 12
2 2 13 13 0 14
1 1 14 14 0 15
4 5 16 16 0 17
```

Ucak id:7 talep edilen saat:2
Kalkis pisti kullanim sirasi

```
3 7 2 2 0 3
2 6 10 10 0 11
2 3 11 11 0 12
3 4 10 12 2 13
2 2 13 13 0 14
1 1 14 14 0 15
4 5 16 16 0 17
```

Ucak id:8 talep edilen saat:14
Kalkis pisti kullanim sirasi

```
3 7 2 2 0 3
2 6 10 10 0 11
2 3 11 11 0 12
3 4 10 12 2 13
2 2 13 13 0 14
1 1 14 14 0 15
3 8 14 15 1 16
4 5 16 16 0 17
```

Ucak id:9 talep edilen saat:17
Kalkis pisti kullanim sirasi

```
3 7 2 2 0 3
2 6 10 10 0 11
```

VIII. KAYNAKÇA

- FOF,Öncelikli kuyruk,Bağlı liste kavramları için;
- [geeksforgeeks.org](https://www.geeksforgeeks.org)
- ocw.mit.edu
- cs.utah.edu/

- Örnek Animasyonlar için;
- brenden.github.io

- codeforces.com
LaTeX Raporu hazırlamak için gerekli ekipman ve bilgiler;
- www.overleaf.com