ÖZET

Proje ile ilk karşılaştığımızda aklımızda gelen sorular vardı. Bunların başında ilk olarak verilen Öncelik kuyruğunu nasıl kullanacağımızı tam olarak anlayamadık. Bu kuyruğu linked list’te mi oluşturacaktık yoksa dizileri mi kullanacaktık emin değildik. Ardından bizden bu kuyruk yapısını kullanarak bir havalimanı iniş kalkış saatlerini uçaklar için otomasyon gibi yazmamız isteniyordu.Daha önce Kuyruk kullanmış olsakta bu isterler ile Öncelik kuyruğunu ortaklaşa kullanmak konusunda net bir fikrimiz yoktu.Her uçak iniş aldıktan sonra ekranda gösterilmesi ve uçakların iniş prosedürlerine ilişkin bir sürü detay ve ister bulunmaktaydı. Bunları nasıl gerçeklemeliyiz bu konuların cevaplarını başta tam olarak bulamadık , araştırmalarımızı yaptıktan sonra gidilebilecek bir yol bulmaya çalıştık zaman harcanıp emek sarf edildikten sonra başarılamayacak kadar zor bir proje olmadığının da farkındaydık. Fakat başlangıçta büyük bir sorun gibi gözüktü çünkü bir kısmı için aklımızda bir yol hesap edebiliyorduk, fakat kesinliğinden emin değildik bir çok hata ile karşılaşabilir hatta sonuca ulaşamayabilirdik. Bu da bizi daha da düşündürüyordu. Bunun ardından çevremizdeki bazı kod bilen insanlardan bu soru işaretlerine verebilecekleri bir cevap veya gösterebilecekleri bir yol var mı öğrenmeye çalıştık ama direkt olarak bizim ihtiyacımız olan bilgiyi veremiyorlardı en sonunda koda giriş yaptık bir kaç hatanın ardından Öncelik kuyruğu hakkında yazılar bulup başkaları nasıl yapmışlar inceledik , Bu uçak iniş planları için öncelik kuyruğu yazıları , ve bir çok diğer algoritmaları inceledik sonunda da anlayıp koda döktük projemizin özgün değeri harcanan zaman yapılan araştırmalar ve incelenen kodlar sebebiyle yüksek olduğunu düşünüyoruz., sonuca ulaşma yani beklenen sonuca ulaşma etkimiz ise %99’dur.

GİRİŞ

Projemizde Öncelikle bizden istenen Öncelik kuyruğu için dizileri ,dosya okuma ,yazma ve diğer bir çok algoritmaları kullanmamızdır. Tabi bunları kullanmamız gereken yerler projede belirtilen soruların çözümlerinde olacaktır . Örnek vermek gerekirse öncelikle iniş yapacak uçaklar kuleden iniş talep etmelidir. İniş izni talep eden her bir uçak için havalimanında yeterli kapasite olup olmadığı kontrol edilmelidir.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 takeoff\_runway\_acceptance\_priority’de uygun yere eklenmelidir. Aksi halde uçakların iniş sıralaması önceliğe göre belirlenmelidir.Uçakların iniş ve kalkış saatleri önceliğe göre belirlenecektir. Gibi bir çok ister bulunmaktadır. Öncelikle İniş pisti öncelik kuyruğu kısmını biz başlangıçta kendi kendimize yazmayı denedik ve yolda katettik fakat sonuca ulaşma kısmında çok fazla zamanımızı aldı . Sonrasında internette yaptığımız araştırmalar eşliğinde bir sitede bir yazı ile karşılaştık tüm araştırdığımız yazılar arasında en anlaşılır olanı ve işimize yarayanı o idi . Orda paylaşılan kodun üzerinde belirli değişiklikler yaptıktan sonra onu kullanmaya karar verdik ve düzgün bir şekilde üzerinde güncelleme işlemlerini gerçekledik. Aynı zamanda Yapmış olduğumuz bu projenin bize göre amaçlarından ilki kod yazımında bizi geliştirerek ilerde meslek sahibi olmamız yolundaki gereken adımları atmamızı sağlamak ve aynı zamanda Programlama konusunda öğrenmiş olduğumuz bilgileri unutmamızı engelleyip hafızamızı dinç tutmak tabii ki bunun yanında farklı amaçlarıda var bizim öğrendiğimiz dosya okuma yazma diziler daha da artırılabilir bilgimizi kullanıp bu bilgilerimizi bir projede bir problemde çözüm olarak nasıl kullanmamız gerektiğini bize öğretmenizdir.

diğer amacı aynı zamanda farklı problemlerle bizi karşılaştırıp farklı düşünme , sorunları çözdürerek kendimizi geliştirmek gibi daha artırılabilir amaçları bulunmaktadır.

Erişilen çıktılar analizlerin tam olarak söylenmesi gerekirse %90’ına doğru cevaplar vermektedir. Aynı zamanda projemiz istenilenşeylerin çoğunu hatasız gerçeklemektedir. Lakin biz projemizin çalıştığından ne kadar emin olsakta bazen gerçek testlerde anlık sıkıntılar çıkabilmektedir.

Yalancı kod

İnput txt’yi oku ;

Sayısal olan satır sayısını al;

Satır sayısını line sayısı değişkenine ata ;

Döngüyü linesayisi +1 olacak şekilde başlat ;

İlk satır hariç diğer satırları 3 diziye al ;

İniş pisti kuyruğu tüm değerlerini 0 eşitle ;

kalkış pisti kuyruğu tüm değerlerini 0 eşitle ;

line sayisi kadar dönecek döngü oluştur ;

her döngü elemanı için add fonksiyonuna dizideki değeri yolla ;

kalkis pisti öncelikli kuyruğunu güncelle ;

kalkis pisti kuyruğunu yazdır ;

scanf ile kullanıcıdan değer al ;

add fonksiyonunda add değeri 24 defadan az geldiyse ilk if’e gir ;

inilmek istenen saatte uçak var mı kontrol et ;

uçak varsa 2 uçağın önceliğini kontrol et ;

yeni gelen uçağın önceliği yüksekse istediği saate inişe izin ver ;

diğer uçakların iniş saatini 1 saat ertele ;

yeni gelen uçağın önceliği düşükse ;

yeni gelen uçağın önceliğinin daha yüksek olduğu uçağı yada boş yeri ara ;

bulduktan sonra oraya yeni uçağı ekle ;

eğer kendinden düşük öncelikli uçağın yerine geçtiyse;

diğer uçakları 1 saat ertele ;

boş yer bulduysa ;

oraya izin al ;

eğer istenilen saatte uçak yoksa izin ver ;

eğer gelen uçak sayısı 24’ü geçtiyse ;

inilmek istenilen yerdeki uçağın önceliğiyle kontrol et ;

yeni gelen uçak daha yüksek önceliğe sahipse ;

düşük öncelikli uçağı sabihaya yolla ;

yeni uçağı oraya ekle ;

eğer yeni gelen uçağın inmek istediği saatteki uçağın önceliği daha yüksekse ;

yeni gelen uçağa izin verme ;

TEMEL BİLGİLER

Projemizi yaparken yani kodumuzu yazarken kullandığımız şeylere değinecek olursak bunlar dev c++ ide’si ve internetten bulduğumuz çeşitli yazılardır.

YÖNTEM

Kodumuzun başlangıcında girilecek olan input.txt dosyasının adı sabit olduğundan kullanıcıdan almak yerine direkt bu isme sahip olan dosyayı okuma modunda açıyoruz.ardından bir while döngüsü eşliğinde dosyanın sonuna kadar sözel olan ilk satırı almayıp diğer kısımları teker teker kaç tane sayısal satır bulunduğu bilgisini alıyoruz.ve toplamda sayısal olan satır sayısını line sayısı isimli değişkende tutuyoruz. Sonrasında tekrardan dosyayı okumaya başlayıp line sayısı kadar dönecek şekilde döngümüzü başlatıyoruz. İçerisindeki 3 sayısal bilgiyi 3 farklı diziye alıyoruz. Sonrasında inecek olanlar öncelik kuyruk dizisinde line sayısı kadar olacak şekilde alan tanımlayıp tüm değerlerine 0 yazıyoruz. Ardından Line sayısı kadar dönecek bir döngüyle her iniş yapmak isteyen uçak bilgisini add isimli bir fonksiyonumuza değerlerimizi yolluyoruz. Öncelikle fonksiyon dışında tanımlanan bir sayaç ile ilk gelen 24 uçak mı değil mi bunun bilgisi tutulmakta eğer 24 uçaktan daha fazla uçak gelirse inmek isteyen bir uçağın öncelik sıralaması ile ineceği saatteki uçağın öncelik sıralaması karşılaştırılıyor eğer daha önceden iniş izni alan uçağın önceliği yeni gelen uçağın önceliğinden düşükse sabiha gökçen havalimanına yönlendiriliyor eğer yüksek ise yeni gelen uçak iniş izni alamamakta eğer sayaç 24 olmazsa yani henüz ilk 24 saatin hepsi dolmamışsa öncelikle yeni gelen uçağın inmek istediği yerde bir uçak izni var mı bunun kontrolü yapılmakta eğer inilmek istenen yerde bir uçak bulunmuyorsa o zaman oraya iniş izni veriliyor .eğer inilmek istenen saate bir uçak izin aldıysa öncelik durumu kontrol ediliyor eğer iniş saati alan uçağın önceliği daha yüksekse insert fonksiyonuna gönderiliyor ve yeni inmek isteyen uçağın istediği saatten sonraki saatlerdeki uçakların öncelikleri kontrol ediliyor eğer daha yüksek önceliği bulunan bir uçak varsa oraya yeni gelen uçak eklenip sonrasındaki uçaklar 1 saat ileri aktarılıyor . Eğer yeni gelenin önceliği inmek istediği saatten daha düşükse o zaman sıralamada inme saati boş olan bir yer bulup oraya iniş izni alınıyor . Her ekleme işlemi bu şekilde gerçekleştirilip ardından anlık olarak iniş öncelik kuyruğunda yapılan değişikler kalkış kuyruğuna istenen şekilde aktarılıyor ve ekrana yazdırılıyor sonrasında bir diğer satır (iniş izni için uçak bilgisi) geliyor her işlemden sonra yapılanı görüntüleyebilmek için bir scanf kullanıyoruz. Enter tuşuna basarak bir ileriki aşamaya geçebiliyor.

SONUÇLAR

Projemiz de gözlemlenen herhangi bir hata bulunmamaktadır.

Bu çalışma sonucunda başarı oranımız %90’a ulaşmıştır. Yine de proje başlangıcında eksik olan yanlarımız vardı. Öncelik kuyruğu gibi noktalar zor gibi gelmişti. Yaptığımız çeşitli araştırmalar sonucunda bunları koda dökmeye çalıştık ve sonuç olarak herhangi bir sorun yaşamadık.

KAYNAKÇA

[Doğrusal Veri Yapıları 4 - Kuyruk (Queue) | by Tolgahan Çepel | Medium](https://medium.com/@tolgahan.cepel/doğrusal-veri-yapıları-4-kuyruk-queue-dcbd07e8ba77)

<http://www.baskent.edu.tr/~tkaracay/etudio/ders/prg/dataStructures/Collections/ClassPriority>

<https://cdn-acikogretim.istanbul.edu.tr/auzefcontent/20_21_Guz/veri_yapilari/7/index.html>

https://stackoverflow.com/questions/8253889/how-to-implement-priority-queue-in-c-programming?rq=1