

| | |
|----------------------------|--|
| Dersin Adı | : Nesneye Yönelik Programlama |
| Proje – 3 | : Havayolu Uçuş Takip Sistemi |
| Verilme Tarihi | : 11.05.2023 |
| Proje Teslim Tarihi | : 28.05.2023 |
| Kod Kontrol Tarihi | : 30 Mayıs Salı ve 1 Haziran Perşembe |

Nesneye yönelik programlama yöntemi ile, bir havayolu şirketinin günlük uçuşlarının takip edilmesini sağlayacak bir uygulama geliştirilmesi istenmektedir. Bu amaçla uçuşlara ilişkin aşağıdaki bilgilere, hem uçuş numaraları hem kalkış yerleri aracılığıyla hızlı erişilebilmesi istenmektedir:

Uçuş numarası : Tamsayı
Kalkış yeri : 20 karakter uzunluğunda string (Latin harflerinden oluşabilecektir.)
Varış yeri : 20 karakter uzunluğunda string (Latin harflerinden oluşabilecektir.)
Kalkış zamanı : 4 karakter uzunluğunda string
Normal koltuk sayısı : Tamsayı
Lüks koltuk sayısı : Tamsayı

Uçuş bilgilerine uçuş numarası aracılığıyla erişmek için, uçuş numarasına göre küçükten büyüğe doğru sıralı olan bir tek-bağlı liste kullanılacaktır. Uçuş bilgilerine kalkış yeri aracılığıyla erişmek için ise kalkış yerine göre (kalkış yeri aynı ise kalkış zamanına göre) sıralı olan bir çift-bağlı liste kullanılacaktır.

Buna göre, aşağıda listelenen isteklerin bir menü aracılığıyla gerçekleştirilmesi istenmektedir:

1) Lüks koltuk içermeyen yeni bir uçuşun eklenmesi

Önce uçuş numarası kullanıcıdan alınmalı ve bu numaralı bir uçuş varsa ekleme yapılmamalıdır, yoksa uçuşa ilişkin diğer bilgiler kullanıcıdan alınarak ekleme işlemi gerçekleştirilmelidir.

2) Lüks koltuk içeren yeni bir uçuşun eklenmesi

Önce uçuş numarası kullanıcıdan alınmalı ve bu numaralı bir uçuş varsa ekleme yapılmamalıdır, yoksa uçuşa ilişkin diğer bilgiler kullanıcıdan alınarak ekleme işlemi gerçekleştirilmelidir.

3) Bir uçuşun lüks koltuk sayısının güncellenmesi

Önce uçuş numarası kullanıcıdan alınmalı, bu numaralı bir uçuş varsa uçuşun lüks koltuk sayısı güncellenmelidir.

4) Bir uçuşun iptal edilmesi

Önce uçuş numarası kullanıcıdan alınmalı, bu numaralı bir uçuş varsa o uçuşa ilişkin bilgiler silinmelidir.

5) Bir uçuşun bilgilerinin görüntülenmesi

Kullanıcıdan bilgilerini görmek istediği uçuşun numarası alınarak, bu numaralı bir uçuş varsa bilgileri görüntülenmelidir.

6) Lüks koltuk içeren uçuşların listelenmesi

Lüks koltuk içeren tüm uçuşların bilgileri, uçuş numaralarına göre artan sırada listelenmelidir.

7) Bir yere olan uçuşların listelenmesi

Kullanıcıdan varış yeri alınarak, bu varış yerine olan uçuşların bilgileri, kalkış yerlerine göre alfabetik sırada listelenmelidir.

8) Bir yerden kalkan uçuşların listelenmesi

Kullanıcıdan kalkış yeri alınarak, bu kalkış yerinden kalkan uçuşların bilgileri, kalkış zamanlarına göre artan sırada listelenmelidir.

9) Bir yerden bir yere olan uçuşların listelenmesi

Kullanıcıdan kalkış ve varış yeri alınarak, bu kalkış yerinden bu varış yerine olan uçuşların bilgileri, kalkış zamanlarına göre azalan sırada listelenmelidir.

Notlar:

1. Kalkış-varış yeri ile ilgili yapılan işlemlerde büyük-küçük harf ayrımı olmamalıdır.
2. Veri girişlerinin doğru bir şekilde yapılacağını varsayınız, hata kontrolü yapmayınız.
3. Arama işlemlerinde, algoritmanızın etkin olmasına dikkat ediniz.
4. Nesneye yönelik programlama ilkeleri açısından, projenin aşağıdaki özelliklere sahip olması beklenmektedir:
 - a. *Ucus*, *Ucus_Numarasina_Gore_Ucus_Listesi*, *Kalkis_Yerine_Gore_Ucus_Listesi* ve *String* sınıfları mutlaka tanımlanmalıdır.
 - b. Sınıfların veri üyeleri *private* olarak belirtilmelidir.
 - c. Değişmeyeceği öngörülen veri üyeleri (diziler hariç), *const* olarak belirtilmelidir.
 - d. Sınıflarda, gerek duyulan oku ve ayarla (*get&set*) fonksiyonları olmalıdır.
 - e. Yapıcı fonksiyonlarda, uygunsa varsayılan argümanlar kullanılmalıdır.
 - f. Sınıflar için *<<* operatörü aşırı yüklenmelidir.
 - g. *String* sınıfı için gerek duyulan karşılaştırma operatörleri (*<*, *>*, *<=*, *>=*, *=* v.b.), üye fonksiyon yöntemiyle aşırı yüklenmelidir.
 - h. Sınıfların arayüzleri (header dosyaları) ve gerçekleştirmeleri (cpp dosyaları) ayrı olmalıdır.

Raporda bulunması gerekenler:

- Kapak (dersin adı, proje numarası ve adı, öğrenci numarası ve ad-soyadı, teslim tarihi)
- Analiz (problemi kendi cümlelerinizle tanımlayıp açıklayınız)
- Tasarım (problemin çözümü için gerekli olan soyut veri tiplerinin (sınıfların) arayüzlerini yazınız, veri üyelerini ve üye fonksiyonlarını açıklayınız)
- Programcı Kataloğu (analiz, tasarım, gerçekleştirim, test ve raporlama için harcadığınız süreleri yazınız, kaynak kodun çıktısını ekleyiniz)
- Kullanıcı Kataloğu (programın kullanım kılavuzunu ekran görüntüleri de kullanarak hazırlayınız, varsa programdaki kısıtlamaları belirtiniz)

DİKKAT EDİLMESİ GEREKEN NOKTALAR:

Projenin Teslim Edilmesi:

1. Proje, 3 kişilik gruplar halinde yapılacaktır.
2. Projenin, adı grup üyelerinin öğrenci numaralarının alt çizgi karakteri ile birleşiminden (örneğin 210710001_220710001_220710002) oluşan **proje klasörü (bin ve obj klasörleri silindikten sonra)** sıkıştırılmalı, sonra da oluşan dosya grup üyelerinden birisi tarafından OBS sisteminde ödev kısmına yüklenmelidir.
3. Rapor, kodla birlikte rar veya zip dosyası yapılarak OBS üzerinden proje yükleme klasörüne birlikte yüklenmelidir. **Online olacak olan kod kontrolü sırasında tüm öğrenciler kendilerine belirtilen saatten en az 15 dakika önce hazır bulunmalıdır.**

Projenin Değerlendirilmesi:

1. Projenin değerlendirmesinde; programın doğru ve eksiksiz çalışmasının yanında etkinlik, nesneye yönelik programlama ilkelerine uygunluk ta dikkate alınacaktır. Buna göre puanlama şu şekildedir:
 - Etkinlik (gereksiz işlemlerden kaçınma): 5 puan
 - NYP ilkelerine uyum: 25 puan
 - Doğru çalışma: 40 puan
 - Rapor: 20 puan
 - Kod kontrolü sırasındaki kişisel performans: 10 puan
2. Kopya çekildiği tespit edildiğinde, çeken ve çektiren kişiler projeden sıfır alacaktır.