**Doğu Ekspresi**

INF212 Project #4

Prepared by

Students ID

1801022035

1801022028

1901022263

Students Name

Ruveyda Dilara Günal

Furkan Küçük

Arif Emre Yıldız

INF 212 Algorithms and Programming II

Spring 2020

Electronic Engineering

Date Submitted: 15.06.2020

Evaluation/Değerlendirme

|  |  |
| --- | --- |
| Report layout /Rapor düzeni | 10 -- |
| Project Flowchart and functions flow chart /Projenin ve fonksiyonların Akış Diyagramı | 25 -- |
| Programming /Programlama Struct, Functions | 45 -- |
| Questions / Sorular | 20 -- |

|  |
| --- |
| Project Objective /Projenin Amacı |
| Doğu Ekspresinde bilet satışı yapacak bir program tasarladık. Normal ve yemekli koltuk türlerine ek olarak belirli koltukları açık artırmayla satıyoruz. Doğu ekspresinde o trenle gitmek isteyip ancak yer bulamayanlar için geliştirilen bir sistem olarak tasarladık. |
| Description of Problem / Problem Tanımlama |
| Doğu Ekspresindeki karaborsa bilet satışlarını engellemek amacıyla isteyenlerin karaborsaya başvurmadan açık arttırmalı sistem ile bilet almalarını sağladık. Böylelikle belirli koltuklarda kişi verdiği teklife göre ya koltuğu alabilecek ya da alamayacak. Koltuğu alabilecek olan müşteriler kendi verdikleri teklifi ödeyecekler. |
| **Description of Method / Metodunun Tanımlaması** |
| Programımızı c++ de yazdık. Bizden istenildiği üzere class yapısını kullandık.  Main fonksiyonundan önce yapmamız gereken her işlemi ayrı ayrı fonksiyonlar şeklinde yazdık. Ilgili işlemleri yerinde ve sırasına uygun şekilde çağırdık. Gerekli durumlarda fonksiyonun parametresi olarak yine fonksiyon verdik. |
| Text of Program/ Programın Kodunu Tanımlaması |
| Kod raporun sonundadır. |
| User’s guide / Kullanıcı Rehberi |
| Bilet almak isteyen kişi kodu çalıştırdıktan sonra öncelikle bilet türünü seçmelidir. Müşteri sonrasında trene biniş ve varış yerini seçerek yolculuğun süresini öğrenir. Yolculuğun kaç saat süreceğini ve güzergahını öğrenen müşteri istenen bilgileri girecektir. İstenen bilgilerdeki yaş alanına girdiği yaşa göre bilet fiyatı indirimli veya indirimsiz olarak kesilecektir. |
| Results of the solution / Programın Sonuçları Program kullanıcıdan yolculuk yapmak istediği vagon türünü seçmesiyle başlıyor. Kullanıcı eğer normal vagonu seçer ise en düşükten ücretlendirecektir. Yemek servisi yoktur. Kullancı bu seçimi yaptıktan sonra nerde bineceğini ve ineceğini seçerek hem güzerhgahı hem de yolculuk süresini görebilmektedir. Yolcudan son olarak kişisel bilgilerini isteyen sistem yolcunun yaşına göre fiyat hesaplamaktadır.  Eğer kullanıcı ilk etapta sorulan vagon türünde yemekli vagonu seçerse normal vagonun iki katı ücretle yolculuk etmeyi seçmektedir. Yemekli vagonda sınırsız yemek bulunmaktadır. Bu vagonda da yaşa göre indirim uygulanmaktadır.  Kullanıcı 3. Seçenek olarak Business vagonu seçerse bilet fiyatları açık artırma ile belirlenecektir. Girilen 6 tekliften en yüksek olan 3 teklifi veren müşteriler koltıklara sahip olacaktır. Ilk önce teklifleri alan program kazanan müşterileri yazdırdıktan sonra sırayla o yolcuların bilgilerini almaktadır. Bu vagonda yaşa göre indirim uygulanmamaktadır. |
| Conclusion and Remarks / Sonuç ve Notlar |
| Programımız istenilen ve bizim planladığımız şekilde çalışıyor.  Vagondaki boş ve dolu yerleri göstermek için yazdığımız kod bize ekstradan algoritma yeteneği kattı ve biraz zorladı. Kodun açık arttırma kısmını yazarken bubble sort diye bilinen algoritmayı kullandık ve pekiştirmiş olduk. Class kullanımına alıştık. Her işlemi ayrı ayrı fonksiyonlarda yaptık yeri geldiğinde fonksiyon parametresi olarak yine fonksiyon kullandık. Bu şekilde fonksiyonlarda özellikle parametre seçiminde kendimizi geliştirmiş olduk.  Projede geliştirilebilecek oldukça fazla alan var. Örneğin açık arttırma algoritması süslenebilir.  Projemiz tamamen kullanıcı dostudur. Kullanıcının kodu çalıştırma şekli oldukça açık ve kolaydır. |
| References / Kaynaklar |
| <https://www.algoritmaornekleri.com/c-plus-plus/c-dizi-elemanlarini-kucukten-buyuge-siralama/>  <https://www.delimuhendis.com/tic-tac-toe-xox/> |

# 

#include<iostream>

using namespace std;

class bilet{

private:

string isim;

string soyisim;

string cinsiyet;

int yas;

int koltuk;

int vagon[4][2]={{0,0},{0,0},{0,0},{0,0}};

int son\_fiyat;

public:

int ankara=0;

int irmak=52;

int kayseri=247;

int sivas=402;

int erzincan=595;

int erzurum=745;

int kars=952;

bilet(string name,string surname,string gender,int age) {

isim=name;

soyisim=surname;

cinsiyet=gender;

yas=age;

}

//~bilet(){}

int biletturu\_sec(){

int secim;

cout<<"Bilet turunu seciniz:"<<endl<<endl;

cout<<"1.Normal"<<endl;

cout<<"2.Yemekli"<<endl;

cout<<"3.Business"<<endl;

cin>>secim;

return secim;

}

int acik\_arttirma(){

int dizi[5],temp;

cout<<"Musteriler tekliflerini girsin:"<<endl;

for(int i=0;i<=5;i++){

cin>>dizi[i];

}

for(int a=0;a<=5;a++){

for(int b=0;b<=5;b++){

if(dizi[b]>dizi[b+1]){

temp=dizi[b];

dizi[b]=dizi[b+1];

dizi[b+1]=temp;

}

}

}

cout<<"Kazanan Musteri Teklifleri:"<<endl;

for(int d=5;d>=3;d--){

cout<<dizi[d]<<endl;

}

}

void veri\_gir(){ //kullanıcı verisi alınıyor

cout<<"\n\nAdiniz : "<<endl;

cin>>isim;

cout<<"Soyadiniz :"<<endl;

cin>>soyisim;

cout<<"Cinsiyetiniz :"<<endl;

cin>>cinsiyet;

cout<<"Yasiniz :"<<endl;

cin>>yas;

}

int kalkis\_yer(){ //kalkıs yeri secimi

int kalkis;

cout<<"0.Ankara"<<endl;

cout<<"1.Irmak"<<endl;

cout<<"2.Kayseri"<<endl;

cout<<"3.Sivas"<<endl;

cout<<"4.Erzincan"<<endl;

cout<<"5.Erzurum"<<endl;

cout<<"6.Kars"<<endl;

cin>>kalkis;

return kalkis;

}

int varis\_yer(){ //varış yeri secimi

int varis;

cout<<"0.Ankara"<<endl;

cout<<"1.Irmak"<<endl;

cout<<"2.Kayseri"<<endl;

cout<<"3.Sivas"<<endl;

cout<<"4.Erzincan"<<endl;

cout<<"5.Erzurum"<<endl;

cout<<"6.Kars"<<endl;

cin>>varis;

return varis;

}

void guzergah(int a,int b){ //guzergah bılgısı

int i;

cout<<"Guzergahiniz: ";

for(i=a ; i<=b ; i++){

if(i==0)

cout<<" Ankara ";

else if(i==1)

cout<<"-> Irmak ";

else if(i==2)

cout<<"-> Kayseri ";

else if(i==3)

cout<<"-> Sivas ";

else if(i==4)

cout<<" -> Erzincan ";

else if(i==5)

cout<<" -> Erzurum ";

else if(i==6)

cout<<" -> Kars ";

}

}

int seyahat\_suresi(int a, int b){ //iki il arası dakika farkını hesaplayan ve onu return eden fonksiyon

int m,h,sum;

sum=b-a;

return sum;

}

void saat\_ve\_dakika(int sum){ //dakika farkını parametre olarak alıp seyahat süresini dk vs saat olarak ayarlayan fonksiyon

int m,h;

m=sum%60;

h=sum/60;

cout<<"\n\nSeyahat sureniz : "<< h <<" saat "<<m<<" dakikadir. "<<endl;

}

int getYas(){ //indirim için privatetaki yaşa bilgisine ulasıyor

return yas;

}

int atama(int b){ //kalkıs/varis yeri fonksiyonlarından return edilen sayı degerlerini şehirele atıyor.

if(b==0){

return ankara;

}

else if (b==1){

return irmak;

}

else if(b==2){

return kayseri;

}

else if(b==3){

return sivas;

}

else if(b==4){

return erzincan;

}

else if(b==5){

return erzurum;

}

else if(b==6);

return kars;

}

void vagon\_goruntule(){ //önce boş atamadan sonra da dolu koltuğu gösteren fonksiyon

for(int i=0;i<4;i++){

for (int j=0;j<2;j++){

if ((j%2)==0){

cout << "\n | ";}

if (vagon[i][j]==0){

cout << (i\*2)+(j+1) << " | ";

}

else if(vagon[i][j]==1){

cout << "X | ";

}

}

}

}

void koltuk\_sec(int koltuk){ //vagon arrayine seçilen koltuğu atayan fonksiyon

if (koltuk<=2){

if(vagon[0][koltuk-1]==0){

cout<<"Yeriniz:"<<koltuk<< endl;

vagon[0][koltuk-1]=1;

}

else{

cout << "\nAlmak istediginiz koltuk doludur.\n";

}

}

else if (koltuk<=4){

if(vagon[1][koltuk-3]==0){

cout<< "Yeriniz:"<<koltuk<< endl;

vagon[1][koltuk-3]=1;

}

else{

cout << "\nAlmak istediginiz koltuk doludur.\n";

}

}

else if (koltuk<=6){

if(vagon[2][koltuk-5]==0){

cout<< "Yeriniz:"<<koltuk<< endl;

vagon[2][koltuk-5]=1;

}

else{

cout << "\nAlmak istediginiz koltuk doludur.\n";

}

}

else if (koltuk<=8){

if(vagon[3][koltuk-7]==0){

cout<< "Yeriniz:"<<koltuk<< endl;

vagon[3][koltuk-7]=1;

}

else{

cout << "\nAlmak istediginiz koltuk doludur.\n";

}

}

}

int fiyat(int sure){ //iki il arasındaki dakika farkına bakarak normal vagonun normal yolcusunun fiyatını hesap. fonks.

int fiyat;

son\_fiyat=sure\*0.08;

return son\_fiyat;

}

void genc\_indirim(int fiyat){ //fiyat donksiyonunu parametre olarak alıp genç indirimi uygulayan fonksiyon

int yeniFiyat;

yeniFiyat=fiyat\*0.7;

son\_fiyat=yeniFiyat;

cout<<"Indirimli Genc Bilet Fiyati ="<<son\_fiyat<<"TL'dir"<<endl;

}

void yasli\_indirim(int fiyat){ //fiyat donksiyonunu parametre olarak alıp yaşlı indirimi uygulayan fonksiyon

int yeniFiyat;

yeniFiyat=fiyat\*0.55;

son\_fiyat=yeniFiyat;

cout<<"Indirimli Yasli Bilet Fiyati ="<<son\_fiyat<<"TL'dir"<<endl;

}

/\* bilet operator+(const bilet& k){

bilet musteri;

musteri.son\_fiyat = this->son\_fiyat+k.son\_fiyat;

return musteri;

}\*/

};

int main(){

string isim,soyisim,cinsiyet;

int a,b,c,yas; //a=bilet türünü atama için gerekli değişken //b ve c= musteriden alınan değerin hangi şehir olduğunu belirten değer

int yemekliVagonFiyat;

int koltuk;

cout<<" \* H O S G E L D I N I Z \* "<<endl;

bilet musteri1(isim,soyisim,cinsiyet,yas); //nesne olusturuldu.

for(int i=1;i<9;i++){

a=musteri1.biletturu\_sec();

cout<<"Kalkis yeri seciniz:"<<endl;

b=musteri1.kalkis\_yer(); //kalkıs yeri secildi.

cout<<"Varis yeri seciniz:"<<endl;

c=musteri1.varis\_yer(); //varis yeri secildi.

musteri1.guzergah(b,c);

musteri1.saat\_ve\_dakika(musteri1.seyahat\_suresi(musteri1.atama(b),musteri1.atama(c)));

cout<<endl<<endl;

if(a==1){ //normal vagon işlemleri

musteri1.vagon\_goruntule(); //vagon görüntülenir.

cout<<"\nKoltuk secimi yapiniz:"<<endl;

cin>>koltuk;

musteri1.koltuk\_sec(koltuk); //girilen parametre vagon arrayine atanır

musteri1.vagon\_goruntule(); //yeni vagon görüntüsü basılır.

musteri1.veri\_gir(); //musteri verileri girilir

if(musteri1.getYas()<=24){

musteri1.genc\_indirim(musteri1.fiyat(musteri1.seyahat\_suresi(musteri1.atama(b),musteri1.atama(c))));} //genç indirimi uygula

else if (musteri1.getYas()>=65){

musteri1.yasli\_indirim(musteri1.fiyat(musteri1.seyahat\_suresi(musteri1.atama(b),musteri1.atama(c))));} //yaşlı indirimi uygula

else{

cout<<"Fiyat: "<<musteri1.fiyat(musteri1.seyahat\_suresi(musteri1.atama(b),musteri1.atama(c)))<<"TL'dir."<<endl;} //indirim yok

}

else if(a==2){ //yemekli vagon işlemleri

musteri1.vagon\_goruntule(); //vagon görüntülenir.

cout<<"\nKoltuk secimi yapiniz:"<<endl;

cin>>koltuk;

musteri1.koltuk\_sec(koltuk); //girilen parametre vagon arrayine atanır

musteri1.vagon\_goruntule(); //yeni vagon görüntüsü basılır.

musteri1.veri\_gir(); //musteri verileri girilir

yemekliVagonFiyat=(musteri1.fiyat(musteri1.seyahat\_suresi(musteri1.atama(b),musteri1.atama(c))))\*2; //normal fiyat\*2

if(musteri1.getYas()<=24){

musteri1.genc\_indirim(yemekliVagonFiyat);} //genç indirimi uygula

else if (musteri1.getYas()>=65){

musteri1.yasli\_indirim(yemekliVagonFiyat);} //yaşlı indirimi uygula

else{

cout<<"Bilet Fiyati: "<<yemekliVagonFiyat<<"TL'dir."<<endl;} //indirim yok

}

else if(a==3){

musteri1.acik\_arttirma();

for(int a=1;a<4;a++){

cout<<a<<". kazanan musteri verileri";

musteri1.veri\_gir();

}

//business vagon işlemleri

}

else{

cout<<"Hatali Giris"<<endl;

}

}

//delete \*must;

return 0;

}