



INF 212 Algorithms and Programming II

**2020-2021 Spring
Electronic Engineering**

Proje ID#
75_

Project Name
P_Simsek

**_Program Name_
*_Bağış Hesap Sistemi_***

Team Members

School ID	Name	Surname
1901022025	Ayşe Serra	Şimşek

PROJECT OBJECTIVE

Briefly describe the objectives of your project

Projenin amacı kısaca yardım kurumlarını, sivil toplum kuruluşlarını, yardımsever kişileri ve de yardıma ihtiyacı olan kişileri ortak bir sistemde bir araya getirip işleri sistemli yürütmek.

PROJECT

Briefly describe the project

İhtiyaç sahiplerine veya sivil toplum kuruluşlarına, çeşitli eşya ve para yardımlarında bulunmak isteyen kişilerin bu otomasyon sistemini kullanarak daha sistemli bir iş yapmaları hedeflenmiştir.

DESIGN ALGORITHM

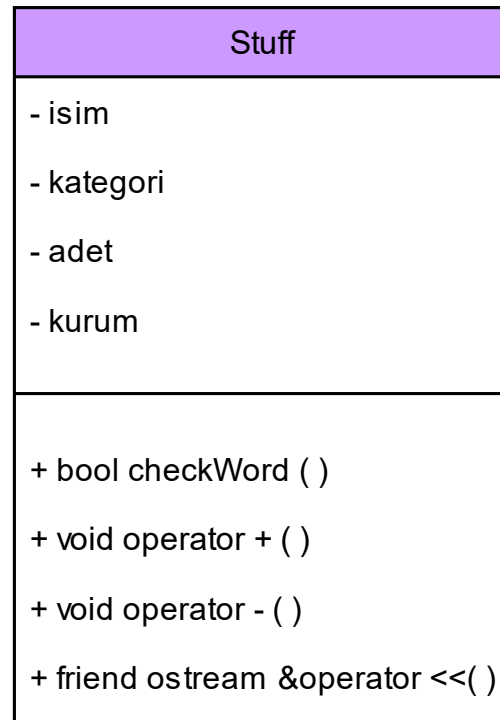
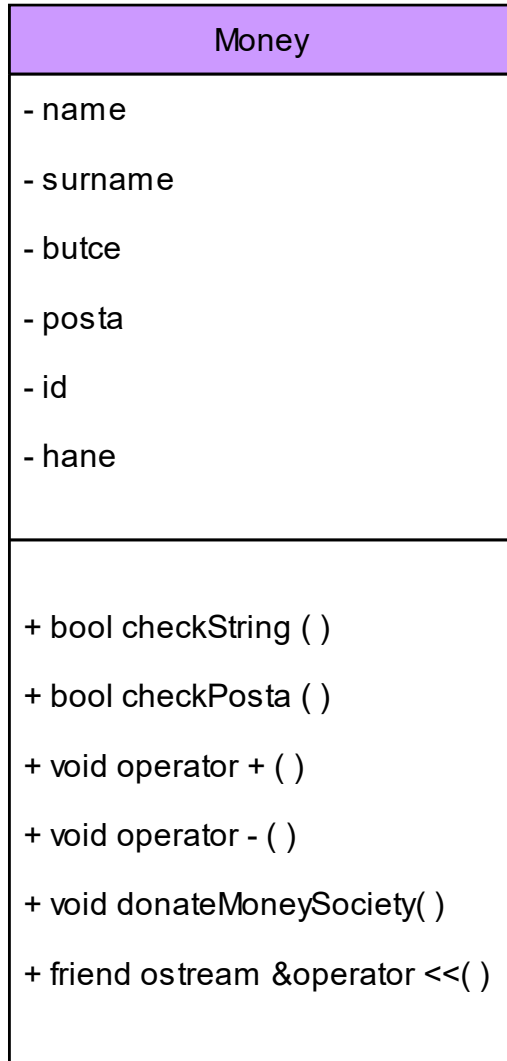
Design algorithm of the project

- Kullanıcı “para bağış”, “eşya bağış” ve “ek üyelik oluşturma ” seçeneklerinden birini seçerek programı kullanmaya başlar.
- Para bağışını seçerse, **bağış almak için, sisteme bağış yapmak için, bir kuruma bağış yapmak için ve işlemi sonlandırmak için** seçenekleriyle karşılaşır. Bu seçeneklerden birini seçerek işlemleri gerçekleştirmeye devam eder.
- Eşya bağışını seçerse, **bağış almak için, sisteme bağış yapmak için, bir kuruma bağış yapmak için ve işlemi sonlandırmak için** seçenekleriyle karşılaşır. Bu seçeneklerden birini seçerek işlemleri gerçekleştirmeye devam eder.
- Ek üyelik oluşturma’yı seçerse gerekli bilgileri girer ve üyelik hesabı, ikinci bir hesaba kopyalanmış olur.

UML DIAGRAM

UML diagram of the project

Classes:



HEADER FILE(s)

Hedeader file(s) of the project

```
#ifndef donation_h_
#define donation_h_
#include<iostream>
#include<string>
using namespace std;

void menu();

class Money{
private:
    string name;
    string surname;
    float butce;
    string posta;
    int *ID;
    int hane;
    friend ostream &operator <<(ostream &out, const Money &obj);

public:
    Money();
    Money(string na, string sur, float but=0, string pho = "", int han=0);
    Money(Money &other);
    ~Money();
    bool checkString(string n);
    bool checkPosta(string posta);

    void operator + (float c);
    void operator - (float w);
    void donateMoneySociety(Money &oth, float miktar);

    void setName(string i);
    void setSurname(string s);
    void setPosta(string p);
    void setButce(float b);
    void setHane(int han);
    void setID(int han);

    int getButce();
    string getPosta();
    string getName();
    string getSurname();
    int getHane();
};
```

```
class Stuff{  
  
private:  
    string isim;  
    int category;  
    int adet;  
    string kurum;  
  
    friend ostream &operator <<(ostream &out, const Stuff &obj);  
  
public:  
    Stuff();  
    Stuff(string i, int c, int a=0, string k="");  
    Stuff(Stuff &other);  
    ~Stuff();  
    bool checkWord(string word);  
  
    void operator + (int al);  
    void operator - (int ver);  
  
    void setIsim(string i);  
    void setCategory(int c);  
    void setAdet(int a);  
    void setKurum(string k);  
  
    int getCategory();  
    string getIsim();  
    int getAdet();  
};  
  
#endif
```

TESTING

Briefly describe how you test your code

- Fonksiyonların parametrelerinin uygunluğuna,
- Class'lara ait fonksiyonları header dosyasında doğru bir şekilde yazmaya,
- Gerekli kütüphaneleri düzgün bir biçimde dahil etmeye,
- If-else / Switch-case / While 'lara ait parantezlere dikkat ettim.

USER'S GUIDE

(Describe briefly how the user can use your program (input parameters of program, output of program, possible limitations, possible errors, etc.)

Kullanıcı programa 3 seçenektan birini seçerek başlar. Seçmiş olduğu seçeneğe göre karşısına tekrar alt seçenekler sunulur. Burada yapmış olduğu seçimlere göre de gerekli bilgileri sisteme girer ve bu şekilde programı kullanmaya başlar.

CONCLUSION AND REMARKS

You can or can(not) put any remarks on your work. For example;
is your program works or not, if not why
which troubles during performance of project you faced with and how you overcome them;
is your program user friendly and how you can improve it;
what you achieved during performance of project; and so

Program doğru çalışıyor. Money class'ında bulunan bir bool fonksiyonunu inheritance ile Stuff class'ında kullanmayı denedim. İstediğim şeyi yapabildim ama bu sefer kodun başka bir yerinin hatalı çalışmasına sebep oldu. Bir süre araştırdım ancak nedenini anlayamadım. Bu sebeple bu şekilde kullanmaktan vazgeçerek bool fonksiyonunu her iki class için ayrı ayrı yazdım.

REFERENCES

Put the list of references and sources (books, links to websites, videos, etc), which you used for project.

- <https://www.udemy.com/share/103GinAEAdcV9RR3gG/>
- www.stackoverflow.com
- www.geeksforgeeks.com

APPENDIX

PROJECT CODES

Put here code of your project.

MAIN.CPP

```
#include "donation.h"
#include <cassert>
#include <time.h>
using namespace std;

int main(){

    srand(time(NULL));

    cout<<"\t\t\t-----";
    cout<<"\n\t\t\t BAGIS HESAP SISTEMI\n";
    cout<<"\t\t\t-----\n\n";

    int choose;
    cout<<"\t\t\t PARA bagis islemleri icin 1\n\t\t\t ESYA bagis islemleri icin 2\n\t\t\t Ek
uyelik olusturmak icin 3\n\t\t\t ";
    cin>>choose;

    if(choose==1){ //PARA

        int secim;
        float cekilen_miktar, yatirilan_miktar, gonderilen_miktar;

        string bir_na, bir_sur, bir_pho;
        int bir_han;
        float bir_but=3000.0;

        string iki_na, iki_sur, iki_pho;
        int iki_han;
        float iki_but=250000.0;

        menu();
        cin>>secim;
        assert(( secim==1 || secim==2 || secim==3 || secim==4 ) );

        Money bir;
        if(secim==1 || secim==2 || secim==3){
```

```
cout<<"\nBilgileri giriniz,"<<endl;
cout<<"Isim:";
    cin>>bir_na;
    bir.setName(bir_na);

cout<<"Soyisim:";
    cin>>bir_sur;
    bir.setSurname(bir_sur);

cout<<"Posta kodu:";
    cin>>bir_pho;
    bir.setPosta(bir_pho);

cout<<"Bagis kartinizda yazan uye no hane sayisi:";
    cin>>bir_han;
    bir.setHane(bir_han);

    bir.setID(bir_han);
    bir.setButce(bir_but);

cout<<bir;
}

while(secim != 4){
    switch(secim){
        case 1:{

            cout<<"Gereken miktari giriniz:";
            cin>>cekilen_miktar;
            bir - cekilen_miktar;
            cout<<"\n--GUNCEL--"<<endl;
            cout<<bir;

            menu();

            cin>>secim;
            assert(( secim==1 || secim==2 || secim==3 || secim==4 ) );
            break;

        }

        case 2:{

            cout<<"\n-----\n";
            cout<<"Sisteme yapacaginiz bagis miktarini giriniz:";
            cin>>yatirilan_miktar;
            bir + yatirilan_miktar;
            cout<<"\n--GUNCEL--"<<endl;
            cout<<bir;

            menu();

            cin>>secim;
```



```
        assert(( secim==1 || secim==2 || secim==3 || secim==4 ) );

        break;

    }

    case 3:{

        Money iki;
        cout<<"\nBagis yapacaginiz kurumun bilgilerini
giriniz,"<<endl;
        cout<<"Kurum adi:";
        cin>>iki_na;
        iki.setName(iki_na);

        cout<<"Kurumun bulundugu il:";
        cin>>iki_sur;
        iki.setSurname(iki_sur);

        cout<<"Posta kodu:";
        cin>>iki_pho;
        iki.setPosta(iki_pho);

        cout<<"Kurum kodunun hane sayisi:";
        cin>>iki_han;

        iki.setID(iki_han);
        iki.setButce(iki_but);

        cout<<"Bagis yapacaginiz miktari giriniz:";
        cin>>gonderilen_miktar;
        bir.donateMoneySociety(iki,gonderilen_miktar);
        cout<<"\n--GUNCEL--"<<endl;
        cout<<bir;
        cout<<iki;

        menu();
        cin>>secim;
        assert(( secim==1 || secim==2 || secim==3 || secim==4 ) );

        break;

    }

    default:{
        cout<<"Hatali giris"<<endl;
        menu();
        cin>>secim;
        assert(( secim==1 || secim==2 || secim==3 || secim==4 ) );

        break;

    }
```

```
        }  
    }  
  
}  
  
if(choose==2){ //ESYA  
  
    int secmek;  
    int cekilen_adet=0, yatirilan_adet=0, gonderilen_adet;  
  
    string first_is, first_kur;  
    int first_cat, first_ad;  
  
    string second_is, second_kur;  
    int second_cat;  
  
    string third_is, third_kur;  
    int third_cat;  
  
    int uye;  
  
    cout<<"\n\t\tUyelik kartinizda bulunan uye numarasini giriniz: ";  
    cin>>uye;  
    cout<<"\t\t"<<uye<<" no'lu uye icin islem baslatildi."<<endl;  
  
    menu();  
    cin>>secmek;  
    assert(( secmek==1 || secmek==2 || secmek==3 || secmek==4 ) );  
  
    while(secmek != 4){  
        switch(secmek){  
            case 1:{  
  
                Stuff first;  
                cout<<first;  
  
                cout<<"\nEsysa bilgilerini giriniz,"<<endl;  
                cout<<"Esysa ismi:";  
                cin>>first_is;  
                first.setIsim(first_is);
```

```
        cout<<"Kategori bilgisi giriniz (kiyafet icin 1, kirtasiye icin 2, mobilya icin 3,
diger 4) :";
        cin>>first_cat;
        first.setCategory(first_cat);

        cout<<"Ihtiyaciniz olan urunun adet bilgisini giriniz:";
        cin>>cekilen_adet;
        first - cekilen_adet;

        cout<<"\n--GUNCEL--"<<endl;
        cout<<first;

        menu();
        cin>>secmek;
        assert(( secmek==1 || secmek==2 || secmek==3 || secmek==4 ) );
        break;
    }

    case 2:{

        Stuff second;
        cout<<"\nEsysa bilgilerini giriniz,"<<endl;
        cout<<"Esysa ismi:";
        cin>>second_is;
        second.setIsim(second_is);

        cout<<"Kategori bilgisi giriniz (kiyafet icin 1, kirtasiye icin 2,
mobilya icin 3, diger 4) :";
        cin>>second_cat;
        second.setCategory(second_cat);

        second.setAdet(72000-cekilen_adet);
        cout<<"Bagislamak istediginiz urunun adet bilgisini giriniz:";
        cin>>yatirilan_adet;
        second + yatirilan_adet;

        cout<<"\n--GUNCEL--"<<endl;
        cout<<second;

        menu();
        cin>>secmek;
        assert(( secmek==1 || secmek==2 || secmek==3 || secmek==4 ) );

        break;
    }

    case 3:{
```

```
Stuff third;
cout<<"\nBagis yapacaginiz kurumun adini giriniz:";
cin>>third_kur;
third.setKurum(third_kur);

cout<<"Esys ismi:";
cin>>third_is;
third.setIsim(third_is);

cout<<"Kategori bilgisi giriniz (kiyafet icin 1, kirtasiye icin 2, mobilya icin 3,
diger 4) :";
cin>>third_cat;
third.setCategory(third_cat);

third.setAdet((72000-cekilen_adet)+yatirilan_adet);
cout<<"Bagislamak istediginiz urunun adet bilgisini giriniz:";
cin>>yatirilan_adet;
third + yatirilan_adet;

cout<<"\n--GUNCEL--"<<endl;
cout<<third;

menu();
cin>>secmek;
assert(( secmek==1 || secmek==2 || secmek==3 || secmek==4 ) );

break;
}

default:{
cout<<"Hatali giris"<<endl;
menu();
cin>>secmek;
assert(( secmek==1 || secmek==2 || secmek==3 || secmek==4 ) );

break;
}

}

}

if(choose==3){ // EK UYELIK (copy constructor)
```

```
Money uye;

string NAME,SURNAME,POSTA;
int hanee;

cout<<"\nBilgileri giriniz,"<<endl;
cout<<"Isim:";
    cin>>NAME;
    uye.setName(NAME);

cout<<"Soyisim:";
    cin>>SURNAME;
    uye.setSurname(SURNAME);

cout<<"Posta kodu:";
    cin>>POSTA;
    uye.setPosta(POSTA);

cout<<"Bagis kartinizda yazan uye no hane sayisi:";
    cin>>hanee;
    uye.setHane(hanee);

    cout<<"\n-----"<<endl;
    cout<<"Isim: "<<uye.getName()<<endl;
    cout<<"Soyisim: "<<uye.getSurname()<<endl;
    cout<<"Posta kodu: "<<uye.getPosta()<<endl;
    cout<<"Uye kartinda bulunan hane sayisi: "<<uye.getHane()<<endl;
    cout<<"-----"<<endl;

Money ekUye(uye);

cout<<"\nVar olan hesabiniza ek uyeliginiz olusturulmustur.(copied)"<<endl;

    cout<<"\n-----"<<endl;
    cout<<"Isim: "<<ekUye.getName()<<endl;
    cout<<"Soyisim: "<<ekUye.getSurname()<<endl;
    cout<<"Posta kodu: "<<ekUye.getPosta()<<endl;
    cout<<"Uye kartinda bulunan hane sayisi: "<<ekUye.getHane()<<endl;
    cout<<"-----"<<endl;

}

return 0;

}
```

DONATION.H

```
#ifndef donation_h_
#define donation_h_
#include<iostream>
#include<string>
using namespace std;

void menu();

class Money{

private:
    string name;
    string surname; // kisi ise soyad, kurum ise sehir bilgisi
    float butce;
    string posta;
    int *ID; //dmm
    int hane;

    friend ostream &operator <<(ostream &out, const Money &obj);

public:
    Money();
    Money(string na, string sur, float but=0, string pho = "", int han=0);
    Money(Money &other);
    ~Money();
    bool checkString(string n);
    bool checkPosta(string posta);

    void operator + (float c);
    void operator - (float w);
    void donateMoneySociety(Money &oth, float miktar);
    void setName(string i);
    void setSurname(string s);
    void setPosta(string p);
    void setButce(float b);
    void setHane(int han);
    void setID(int han);

    int getButce();
    string getPosta();
    string getName();
    string getSurname();
    int getHane();

};
```

```
class Stuff{

private:
    string isim;
    int category; //kiyafet=1, kirtasiye=2, mobilya=3, diger=4
    int adet;
    string kurum;

    friend ostream &operator <<(ostream &out, const Stuff &obj);

public:
    Stuff();
    Stuff(string i, int c, int a=0, string k="");
    Stuff(Stuff &other);
    ~Stuff();
    bool checkWord(string word);

    void operator + (int al);
    void operator - (int ver);

    void setIsim(string i);
    void setCategory(int c);
    void setAdet(int a);
    void setKurum(string k);

    int getCategory();
    string getIsim();
    int getAdet();

};

#endif
```

DONATION.CPP

```
#include "donation.h"
using namespace std;
```

```
void menu(){
    cout<<"\n\nBagis almak icin 1\nSisteme bagis yapmak icin 2\nBir kuruma bagis
gondermek icin 3\nIslemi bitirmek icin 4:\t\t";
}
```

```
Money::Money(){
    name="-";
    surname="-";
    butce=0;
    posta="-";
    hane=0;
    setID(hane);
}
```

```
Money::Money(string na, string sur, float but, string p, int han){

    setName(na);
    setSurname(sur);
    setButce(but);
    setPosta(p);
    setHane(han);
    setID(han);

}
```

```
Money::~~Money(){
    delete [] ID;
}
```

```
Money::Money(Money &other){

    name=other.name;
    surname=other.surname;
    butce=other.butce;
    posta=other.posta;
    hane=other.hane;
```



```
        ID = new int[hane];
        for(int i=0; i<hane; ++i){
            ID[i] = other.ID[i];
        }
    }

void Money::setName(string i){
    while( !checkString(i)){
        cout<<"Gecersiz isim girisi, tekrar deneyiniz:";
        cin>>i;
    }
    name=i;
}

void Money::setSurname(string s){
    while( !checkString(s)){
        cout<<"\nGecersiz soyisim girisi, tekrar deneyiniz:";
        cin>>s;
    }
    surname=s;
}

void Money::setButce(float b){
    if (b <= 0){
        butce=0;
    }
    else{
        butce=b;
    }
}

void Money::setPosta(string p){
    while( !checkPosta(p)){
        cout<<"Gecersiz, posta kodunu dogru hanede giriniz:";
        cin>>p;
    }
    posta=p;
}
```

```
void Money::setHane(int han){
    hane=han;
}

void Money::setID(int han){
    hane=han;
    ID = new int[hane]; //dynamic memory kullanımı
    for(int i=0; i<hane; ++i){
        ID[i]=rand()%9;
    }
}

int Money::getButce(){
    return butce;
}

string Money::getPosta(){
    return posta;
}

string Money::getName(){
    return name;
}

string Money::getSurname(){
    return surname;
}

int Money::getHane(){
    return hane;
}

bool Money::checkString(string n){
    for(int check=0; check<n.size(); check++){
        if(!((n[check] >='A' && n[check] <='Z') || (n[check] >='a' && n[check]
<='z'))){
            return false;
        }
    }
    return true;
}
```

```
void Money::operator + (float miktar){
    while(miktar <= 0){
        cout<<"Gecersiz miktar, lutfen gecerli miktar giriniz:";
        cin>>miktar;
    }
    butce += miktar;
}

void Money::operator - (float miktar){
    while(miktar <= 0 || miktar > butce){
        cout<<"Yeterli miktar yok, lutfen gecerli miktar giriniz:";
        cin>>miktar;
    }
    butce -= miktar;
}

bool Money::checkPosta(string posta){ // posta kodunun 5 haneli olmasi kontrolu

    if( posta.size() != 5){ //size metodu
        return false;
    }

    return true;
}

ostream &operator <<(ostream &out, const Money &obj){ //friend

    out<<"-----"<<endl
    <<"->"<<obj.name<<endl
    <<"->"<<obj.surname<<endl
    <<"->Posta kodu:"<<obj.posta<<endl
    <<"->Fonda bulunan toplam miktar:"<<obj.butce<<" TL"<<endl
    <<"->Gecici ID:";
    for(int i=0; i<obj.hane; ++i){
        out<<obj.ID[i]<<" ";
    }
    out<<"\n-----"<<endl;
    return out;
}
```

```
void Money::donateMoneySociety(Money &oth, float miktar){
    while(miktar <= 0 || butce < miktar){
        cout<<"Fonda yeterli miktar yok, gecerli miktar girmelisiniz:\t"<<endl;
        cin>>miktar;
    }

    butce -= miktar;
    oth.butce += miktar;
}

//-----
Stuff::Stuff(){
    isim="URUNLER";
    category=0;
    adet=72000;
    kurum="TUM KURUMLAR";
}

Stuff::Stuff(string i, int c, int a, string k){

    setIsim(i);
    setCategory(c);
    setAdet(a);
    setKurum(k);
}

Stuff::~~Stuff(){

}

Stuff::Stuff(Stuff &other){

    isim=other.isim;
    category=other.category;
    adet=other.category;
}

bool Stuff::checkWord(string word){
    for(int i=0; i<word.size(); i++){
        if(!(word[i] >='A' && word[i] <='Z') || (word[i] >='a' && word[i] <='z'))){
            return false;
        }
    }
    return true;
}
```

```
void Stuff::operator + (int al){

    while(al <= 0){
        cout<<"Gecersiz adet, lutfen gecerli sayi giriniz:";
        cin>>al;
    }
    adet += al;
}

void Stuff::operator - (int ver){
    while(ver <= 0 || ver > adet ){
        cout<<"Gecersiz adet, lutfen gecerli giris yapiniz:";
        cin>>ver;
    }
    adet -= ver;
}

ostream &operator <<(ostream &out, const Stuff &obj){ //friend

    out<<"-----"<<endl
    <<"->Urun ismi:"<<obj.isim<<endl
    <<"->Kategori:"<<obj.category<<endl
    <<"->Fonda bulunan esya:"<<obj.adet<<" adet"<<endl
    <<"->Kurum:"<<obj.kurum<<endl;

    out<<"-----"<<endl;
    return out;
}

void Stuff::setIsim(string i){
    while( !checkWord(i)){
        cout<<"Gecersiz isim, tekrar deneyiniz:";
        cin>>i;
    }
    isim=i;
}

void Stuff::setKurum(string k){
    while( !checkWord(k)){
        cout<<"Gecersiz isim, tekrar deneyiniz:";
        cin>>k;
    }
}
```

```
        kurum=k;
    }

    void Stuff::setCategory(int c){

        category=c;
    }

    void Stuff::setAdet(int a){

        adet=a;
    }

    int Stuff::getCategory(){
        return category;
    }

    string Stuff::getIsim(){
        return isim;
    }

    int Stuff::getAdet(){
        return adet;
    }
}
```