# Бүрдэл Харьцааны Лабораторийн Ажил (Лаборатори №8)

# 1. ОРШИЛ/УДИРТГАЛ

Уг лабораторын ажлаар класс хоорондын удамшил болон бүрдэл харьцааны тусламжтайгаар хэд хэдэн класс тодорхойлж боловсруулалт хийсэн. Векторын талаар ойлголттой болсон,

# 2. ЗОРИЛГО

Оюутан нэртэй объектын классыг өгөгдсөн шаардлагын дагуу зарлаж, гишүүн функцуудийг тодорхойлно. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Класс диаграмын дагуу классуудаг байгуулах,
2. Класс бүрт байгуулагч тодорхойлох
3. Employee классын объект тус бүрд Spouse, Child – уудыг тохируулж өгөх,
4. Employee классын объект тус бүрийн бүх мэдээллийг хэвлэх,
5. 0..1, 0..n болон 1, 1..n харьцаануудыг зөв програмчлах.

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 Удамшлын харьцаа

Удамшиж буй болон удамшуулж буй классуудын хооронд “is a”, “is a kind of” буюу тэр бол тэр, тэр бол тэр төрлийнх гэсэн харьцаа үүсдэг. Жишээ нь хүн классаас удамшсан сурагч классын хооронд сурагч бол хүн гэсэн харьцаа харагдаж байна. Удамшил нь өөр нэг классын гишүүн өгөгдлүүдтэй адил гишүүдтэй классыг тодорхойлоход хэсэг кодыг дахин бичихгүйн тулд хэрэглэнэ.

## 3.2 Бүрдэл харьцаа

Ямар нэг классд өөр нэг классын объект гишүүн өгөгдөл болохыг бүрдэл гэнэ. Бүрдэл нь “has a” буюу ийм юмтай гэсэн харьцааг үүсгэнэ. Жишээ нь машин нь хөдөлгүүртэй буюу машин класс нь хөдөлгүүр классын объектыг агуулж байна.

## 3.3 Бүрдэл харьцааны хориглолт

Классууд тодорхой зааг буюу хориглолттойгоор хоорондоо харьцааг үүсгэдэг. Үүнд:

* 1...1 харьцаа нь класс нөгөө классын нэг объектыг агуулна гэсэн үг. Жишээ нь хүн бүр нэг л төрсөн ээжтэй гэх мэт.
* 0...1 харьцаа нь класс нөгөө классын нэгээс дээшгүй объектыг агуулна гэсэн үг. Жишээ нь хүнд 1с олон эхнэр\нөхөр байж болохгүй гэх мэт.
* 0...n харьцаа нь класс нөгөө классын хэдэн ч объектыг агуулж болно гэсэн үг. Жишээ нь хүнд хэдэн ч сайн найз байж болно гэх мэт.
* 1…n харьцаа нь класс нөгөө классын дор хаяж нэг объектыг агуулна гэсэн үг. Жишээ нь тэмцээн ядаж нэг ялагчтай байна гэх мэт.

# 4. ХЭРЭГЖҮҮЛЭЛТ

Хамгийн эхлээд классуудыг класс диаграммын дагуу тодорхойлсон.

class Person {

protected:

    char name[20];

    char ssnum[20];

    int age;

public:

    Person(char a[]="ner", char b[]="register", int c=0)

        {strcpy(name, a); strcpy(ssnum, b); age=c;}

    char\* getName()

        {return name;}

    char\* getSSNum()

        {return ssnum;}

    int getAge()

        {return age;}

    void setName(char\* a)

        {strcpy(name, a);}

    void setSSNum(char\* a)

        {strcpy(ssnum, a);}

    void setAge(int a)

        {age = a;}

};

class Spouse:public Person {

protected:

    int anniversaryYear;

public:

    Spouse(char a[]="ner", char b[]="register", int c=2022, int d=2022)

        {strcpy(name, a);

        strcpy(ssnum, b);

        age = c;

        anniversaryYear = d;}

    int getAnniversaryYear()

        {return anniversaryYear;}

    void setAnniversaryYear(int a)

        {anniversaryYear = a;}

};

class Child:public Person {

protected:

    char favoriteToy[20];

public:

    Child(char a[]="ner", char b[]="register", int c=2022, char d[]="togloom")

        {strcpy(name, a);

        strcpy(ssnum, b);

        age = c;

        strcpy(favoriteToy, d);}

    char\* getFavoriteToy()

        {return favoriteToy;}

    void setFavoriteToy(char\* a)

        {strcpy(favoriteToy, a);}

};

class Division {

protected:

    char divisionName[20];

public:

    Division(char a[]="heltes")

        {strcpy(divisionName, a);}

    char\* getDivisionName()

        {return divisionName;}

    void setDivisionName(char\* a)

        {strcpy(divisionName, a);}

};

class JobDescription {

protected:

    char description[20];

public:

    JobDescription(char a[]="tailbar")

        {strcpy(description, a);}

    char\* getDescription()

        {return description;}

    void setDescription(char\* a)

        {strcpy(description, a);}

};

class Employee:public Person {

protected:

    vector<Spouse> hani;

    vector<Child> huuhed;

    Division heltes;

    vector<JobDescription> todorhoilolt;

    char companyId[20];

    char title[20];

    int startYear;

public:

    Employee(char a[]="ner", char b[]="register", int c=2022, char d[]="company", char e[]="title", int f=2022)

        {strcpy(name, a);

        strcpy(ssnum, b);

        age = c;

        strcpy(companyId, d);

        strcpy(title, e);

        startYear = f;

        JobDescription k;

        todorhoilolt.push\_back(k);}

    char\* getCompanyId()

        {return companyId;}

    char\* getTitle()

        {return title;}

    int getStartYear()

        {return startYear;}

    void setCompanyId(char\* a)

        {strcpy(companyId, a);}

    void setTitle(char\* a)

        {strcpy(title, a);}

    void setStartYear(int a)

        {startYear = a;}

    char\* getDivisionName()

        {return heltes.getDivisionName();}

    void setDivisionName(char\* a)

        {heltes.setDivisionName(a);}

    void addSpouse() {

        if(hani.size()<2) {

            int jil, nas;

            char ner[20], rd[20];

            cout << "Ger buliin hunii ner:";

            cin >> ner;

            cout << "Ger buliin hunii register:";

            cin >> rd;

            cout << "Ger buliin hunii nas:";

            cin >> nas;

            cout << "Gerlesen jil:";

            cin >> jil;

            Spouse a(ner,rd,nas,jil);

            hani.push\_back(a);

        }

    }

    void addChild() {

        char ner[20], rd[20], togloom[20];

        int nas;

        cout << "Huuhdiin ner:";

        cin >> ner;

        cout << "Huuhdiin register:";

        cin >> rd;

        cout << "Huuhdiin nas:";

        cin >> nas;

        cout << "Huuhdiin durtai togloom:";

        cin >> togloom;

        Child a(ner, rd, nas, togloom);

        huuhed.push\_back(a);

    }

    void addJobDescription() {

        char tailbar[50];

        cout << "Ajliin todorhoiloltiig oruul:";

        cin >> tailbar;

        JobDescription a(tailbar);

        todorhoilolt.push\_back(a);

    }

    void print() {

        cout << "Ajiltan " << name << " ni ";

        cout << ssnum << " registertei, ";

        cout << age << " nastai, ";

        cout << companyId << " dugaartai bogood, ";

        cout << startYear << " onoos ";

        cout << heltes.getDivisionName() << " heltest ";

        cout << title << " alban tushaaltai.";

        for(int i=0; i<hani.size(); i++) {

            cout << "\n\tGer buliin hun:";

            cout << "ner:" << hani[0].getName();

            cout << ", register:" << hani[0].getSSNum();

            cout << ", nas:" << hani[0].getAge();

            cout << ", gerlesen jil:" << hani[0].getAnniversaryYear();

        }

        for(int i=0; i<huuhed.size(); i++) {

            cout << "\n\tHuuhed: ";

            cout << "ner:" << huuhed[i].getName();

            cout << ", register:" << huuhed[i].getSSNum();

            cout << ", nas:" << huuhed[i].getAge();

            cout << ", durtai togloom:" << huuhed[i].getFavoriteToy();

        }

        for(int i=0; i<todorhoilolt.size(); i++) {

            cout << "\n\tTodorhoilolt " << i+1 << ": ";

            cout << todorhoilolt[i].getDescription();

        }

    }

};

тэгээд main функц дээр ажилчид векторыг үүсгээд ажилчдыг нэмж бүртгэн бусад объектон өгөгдлүүдийн гишүүдийг оруулдаг байхаар давталтаар зохион байгуулсан.

# 5. ДҮГНЭЛТ

Бүрдлийг ашигласнаар кодыг дахин бичихээс сэргийлж байна. Мөн гишүүн өгөгдлүүдийг илүү логиктой зохион байгуулах боломжийг олгож байна. Жишээ нь бүрдлийг ашиглахгүйгээр машин классыг тодорхойлвол машины тодорын шинж чанарыг бичих гишүүн өгөгдлүүд тодорхойлох шаардлагатай болно. Эсвэл тэр хэсэг кодыг удамшлын тусламжтайгаар хэрэглэж болох хэдий ч мотороос машин удамшина гэдэг нь логикын хувьд алдаатай болж байна. Тиймээс энэ үед бүрдэл хэрэг болно.

# 6. АШИГЛАСАН МАТЕРИАЛ

1. <https://www.geeksforgeeks.org/object-composition-delegation-in-c-with-examples/>

2. <https://www.geeksforgeeks.org/inheritance-in-c/?ref=lbp>

# 7. ХАВСРАЛТ

#include <vector>

#include <iostream>

#include <string.h>

using namespace std;

class Person {

protected:

    char name[20];

    char ssnum[20];

    int age;

public:

    Person(char a[]="ner", char b[]="register", int c=0)

        {strcpy(name, a); strcpy(ssnum, b); age=c;}

    char\* getName()

        {return name;}

    char\* getSSNum()

        {return ssnum;}

    int getAge()

        {return age;}

    void setName(char\* a)

        {strcpy(name, a);}

    void setSSNum(char\* a)

        {strcpy(ssnum, a);}

    void setAge(int a)

        {age = a;}

};

class Spouse:public Person {

protected:

    int anniversaryYear;

public:

    Spouse(char a[]="ner", char b[]="register", int c=2022, int d=2022)

        {strcpy(name, a);

        strcpy(ssnum, b);

        age = c;

        anniversaryYear = d;}

    int getAnniversaryYear()

        {return anniversaryYear;}

    void setAnniversaryYear(int a)

        {anniversaryYear = a;}

};

class Child:public Person {

protected:

    char favoriteToy[20];

public:

    Child(char a[]="ner", char b[]="register", int c=2022, char d[]="togloom")

        {strcpy(name, a);

        strcpy(ssnum, b);

        age = c;

        strcpy(favoriteToy, d);}

    char\* getFavoriteToy()

        {return favoriteToy;}

    void setFavoriteToy(char\* a)

        {strcpy(favoriteToy, a);}

};

class Division {

protected:

    char divisionName[20];

public:

    Division(char a[]="heltes")

        {strcpy(divisionName, a);}

    char\* getDivisionName()

        {return divisionName;}

    void setDivisionName(char\* a)

        {strcpy(divisionName, a);}

};

class JobDescription {

protected:

    char description[20];

public:

    JobDescription(char a[]="tailbar")

        {strcpy(description, a);}

    char\* getDescription()

        {return description;}

    void setDescription(char\* a)

        {strcpy(description, a);}

};

class Employee:public Person {

protected:

    vector<Spouse> hani;

    vector<Child> huuhed;

    Division heltes;

    vector<JobDescription> todorhoilolt;

    char companyId[20];

    char title[20];

    int startYear;

public:

    Employee(char a[]="ner", char b[]="register", int c=2022, char d[]="company", char e[]="title", int f=2022)

        {strcpy(name, a);

        strcpy(ssnum, b);

        age = c;

        strcpy(companyId, d);

        strcpy(title, e);

        startYear = f;

        JobDescription k;

        todorhoilolt.push\_back(k);}

    char\* getCompanyId()

        {return companyId;}

    char\* getTitle()

        {return title;}

    int getStartYear()

        {return startYear;}

    void setCompanyId(char\* a)

        {strcpy(companyId, a);}

    void setTitle(char\* a)

        {strcpy(title, a);}

    void setStartYear(int a)

        {startYear = a;}

    char\* getDivisionName()

        {return heltes.getDivisionName();}

    void setDivisionName(char\* a)

        {heltes.setDivisionName(a);}

    void addSpouse() {

        if(hani.size()<2) {

            int jil, nas;

            char ner[20], rd[20];

            cout << "Ger buliin hunii ner:";

            cin >> ner;

            cout << "Ger buliin hunii register:";

            cin >> rd;

            cout << "Ger buliin hunii nas:";

            cin >> nas;

            cout << "Gerlesen jil:";

            cin >> jil;

            Spouse a(ner,rd,nas,jil);

            hani.push\_back(a);

        }

    }

    void addChild() {

        char ner[20], rd[20], togloom[20];

        int nas;

        cout << "Huuhdiin ner:";

        cin >> ner;

        cout << "Huuhdiin register:";

        cin >> rd;

        cout << "Huuhdiin nas:";

        cin >> nas;

        cout << "Huuhdiin durtai togloom:";

        cin >> togloom;

        Child a(ner, rd, nas, togloom);

        huuhed.push\_back(a);

    }

    void addJobDescription() {

        char tailbar[50];

        cout << "Ajliin todorhoiloltiig oruul:";

        cin >> tailbar;

        JobDescription a(tailbar);

        todorhoilolt.push\_back(a);

    }

    void print() {

        cout << "Ajiltan " << name << " ni ";

        cout << ssnum << " registertei, ";

        cout << age << " nastai, ";

        cout << companyId << " dugaartai bogood, ";

        cout << startYear << " onoos ";

        cout << heltes.getDivisionName() << " heltest ";

        cout << title << " alban tushaaltai.";

        for(int i=0; i<hani.size(); i++) {

            cout << "\n\tGer buliin hun:";

            cout << "ner:" << hani[0].getName();

            cout << ", register:" << hani[0].getSSNum();

            cout << ", nas:" << hani[0].getAge();

            cout << ", gerlesen jil:" << hani[0].getAnniversaryYear();

        }

        for(int i=0; i<huuhed.size(); i++) {

            cout << "\n\tHuuhed: ";

            cout << "ner:" << huuhed[i].getName();

            cout << ", register:" << huuhed[i].getSSNum();

            cout << ", nas:" << huuhed[i].getAge();

            cout << ", durtai togloom:" << huuhed[i].getFavoriteToy();

        }

        for(int i=0; i<todorhoilolt.size(); i++) {

            cout << "\n\tTodorhoilolt " << i+1 << ": ";

            cout << todorhoilolt[i].getDescription();

        }

    }

};

int main() {

    int k;

    vector<Employee> ajilchid;

    while(true) {

        cout << "1.Ajiltan nemeh\n";

        cout << "2.Ajiltnii ger buliin gishuu medeelliig nemeh\n";

        cout << "3.Ajiltnii huuhdiin medeelliig nemeh\n";

        cout << "4.Ajiltnii heltsiig oorchloh\n";

        cout << "5.Ajiltnii ajliin todorhoilolt nemeh\n";

        cout << "6.Ajilchdiin medeelliig haruulah\n";

        cout << "7.Duusgah\n";

        cin >> k;

        if(k==1) {

            char ner[20], rd[20], comId[20], title[20];

            int nas, ajildOrsonJil;

            cout << "Ajiltnii ner:";

            cin >> ner;

            cout << "Ajiltnii register:";

            cin >> rd;

            cout << "Ajiltnii nas:";

            cin >> nas;

            cout << "Ajiltnii dugaar:";

            cin >> comId;

            cout << "Ajiltnii alban tushaal:";

            cin >> title;

            cout << "Ajiltnii ajilj ehelsen on:";

            cin >> ajildOrsonJil;

            Employee ajilchin(ner, rd, nas, comId, title, ajildOrsonJil);

            ajilchid.push\_back(ajilchin);

        } if(k==2) {

            char id[20];

            int p=0;

            cout << "Ger buliin gishuunii medeelel nemeh ajiltnii dugaariig oruul:";

            cin >> id;

            for(int i=0; i<ajilchid.size(); i++) {

                if(strcmp(id, ajilchid[i].getCompanyId()) == 0) {

                    ajilchid[i].addSpouse();

                    p++;

                }

            }

            if(p==0)

                cout << "Ug dugaartai ajiltan oldsongui.\n";

        } if(k==3) {

            char id[20];

            int p=0;

            cout << "Huuhdiin medeelel nemeh ajiltnii dugaariig oruul:";

            cin >> id;

            for(int i=0; i<ajilchid.size(); i++) {

                if(strcmp(id, ajilchid[i].getCompanyId()) == 0) {

                    ajilchid[i].addChild();

                    p++;

                }

            }

            if(p==0)

                cout << "Ug dugaartai ajiltan oldsongui.\n";

        } if(k==4) {

            char id[20];

            int p=0;

            cout << "Tasgiin neriig oorchloh ajiltnii dugaariig oruul:";

            cin >> id;

            for(int i=0; i<ajilchid.size(); i++) {

                if(strcmp(id, ajilchid[i].getCompanyId()) == 0) {

                    char tasag[20];

                    cout << "Tasgiin neriig oruul:";

                    cin >> tasag;

                    ajilchid[i].setDivisionName(tasag);

                    p++;

                }

            }

            if(p==0)

                cout << "Ug dugaartai ajiltan oldsongui.\n";

        } if(k==5) {

            char id[20];

            int p=0;

            cout << "Ajliin todorhoilolt nemeh ajiltnii dugaariig oruul:";

            cin >> id;

            for(int i=0; i<ajilchid.size(); i++) {

                if(strcmp(id, ajilchid[i].getCompanyId()) == 0) {

                    ajilchid[i].addJobDescription();

                    p++;

                }

            }

            if(p==0)

                cout << "Ug dugaartai ajiltan oldsongui.\n";

        } if(k==6) {

            for(int i=0; i<ajilchid.size(); i++) {

                ajilchid[i].print();

            }

        } if(k==7) {

            break;

        }

    }

}