# Class-д private хандалтын түвшинтэй хаяган хувьсагч зарлаж түүнд хандах лабораторийн ажил (Лаборатори №5)

М.Солонго

ХШУИС, МКУТ-ийн оюутан, 20b1num2104@stud.num.edu.mn

# 1. ОРШИЛ

Ажилчин класс зарлаж түүндээ гишүүн өгөгдөл, гишүүн функц, байгуулагч, устгагч, хуулагч функц тодорхойлж ID өмнө нь бүртгэгдсэн эсэхийг шалган, хуулагч функц болон хаягийн тусламжтайгаар объектыг нэрээр нь эрэмбэлэхийн тулд зорилго тодорхойлж код бичин холбогдох байгуулагч, хуулагч, устгагч функцийн онолыг тус тус судлан хавсаргав.

# 2. ЗОРИЛГО

Private хандалтын түвшинтэй хаяган хувьсагч класс дотор тодорхойлох мөн объектын хаяган хүснэгт үүсгээд хүснэгтээ нэрээр нь эрэмбэлэхийн тулд дараах зорилтуудыг тавьж ажилласан:

1. Лекцийн файлыг уншиж судлах
2. Объект хандлагат програмчлалын номноос холбогдох сэдвийг унших
3. Багшийн хавсаргасан кодонд дүн шинжилгээ хийх
4. Хаяган хувьсагч зарлаж түүндээ get set функц бичиж, ашиглах
5. Хаяган хүснэгт тодорхойлон нэрээр нь эрэмбэлэх

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 Байгуулагч функц

Анхдагч байгуулагч тодорхойлж өгөөгүй тохиолдолд compiler функцийг далдуур нэмэлтээр тодорхойлдог ажилуулдаг. Параметртэй байгуулагч нь гишүүн өгөгдөлд утга оноох шаардлагагүй. Дамжуулан өгөх утгыг гаднаас авч болдгоороо давуу талтай.

## 3.2 Устгагч функц

Устгагч функцгүйгээр байгуулсан объектуудыг устгаж чадахгүй. Тодорхойлж өгөөгүй тохиолдолд compiler нэмэлтээр анхдагч байгуулагчийн адил далдуур тодорхойлж өгдөг.

# 4. ХЭРЭГЖҮҮЛЭЛТ

## 4.1 Байгуулагч функц хэзээ дуудагддаг вэ?

Анхдагч байгуулагч объект зарлах үед дуудагдана.

## 4.2 Устгагч функц хэзээ дуудагдах вэ?

Програм ажиллаж дуусахаас өмнө дуудагдаж new оператороор нөөцлөгдсөн санах ойг чөлөөлнө.

Харин delete оператор багц командын хаалт дуусахаас өмнө санах ойг чөлөөлнө.

## 4.3 Хуулагч байгуулагч гэж юу вэ? Ач холбогдол юу вэ?

Хуулагч байгуулагч нь түүний өмнөх аль нэг объектын сүүлийн утгыг хуулбарлаж авах боломжийг олгох функц.

Бичихдээ:

Employee(employee& emp){

Strcpy(name,emp.name);

Tsalin=emp.tsalin;

Ajillasan\_tsag=emp.ajillasan\_tsag;

}

## 4.4 Хуулагч функц гэж юу вэ? Ач холбогдол юу вэ? Санах ойн цоорхойгоос хэрхэн сэргийлэх вэ?

Main функц дотор класс төрөлтэй объект тодорхойлоод ямар нэгэн байрыг нь солих хэрэгцээ шаардлага тулгарсан тохиолдолд хуулагч функц хэрэглэдэг. Энэ нь ойн цоорхой үүсэхээс сэргийлж delete new операторын тусламжтайгаар 2 объектын байрыг солих үйлдэл хийнэ.

## 4.5 Объектын хаяган хувьсагчийг хэрхэн зарлах вэ? New оператороор санах ой нөөцлөх, хаяган хувьсагчаар дамжуулаж объектын гишүүн өгөгдөл, гишүүн функцэд яаж хандах вэ?

Зарлахдаа:

employee \*obj1;

Хаяган хувьсагчаар дамжуулж объектын гишүүн өгөгдөл, гишүүн функцд хандах:

Cout<<obj1->get\_name;

Cout<<obj1->get\_ajillasan\_tsag

Үүнд name, ajillasan\_tsag гишүүн өгөгдөл нь private хандалтын түвшинтэй учир get функцээр дамжуулан хандаж байна. Public гишүүн өгөгдөлтэй тохиолдолд функцээр дамжуулалгүй шууд хандаж болно.

Cout<<obj1->number;

## 4.6 Объектын хаяган хувьсагчийн хүснэгт хэрхэн үүсгэх вэ? Жишээ код бичиж ажилуул

# 5. ДҮГНЭЛТ

Хаяган хүснэгт үүсгэн хаягаар нь эрэмбэлэх нь объектын байрыг сольж эрэмбэлэхээс илүүтэйгээр эрсдэл бага. Объектын байрыг сольж эрэмбэлэхэд код санах ой ихээр үүсгэж, устгаж байгаатай холбоотойгоор бичиж буй хүний санамсаргүй алдаанаас санах ойн цоорхой үүсэх, segmentation fault үүсгэх эрсдэлтэй.

# 6. АШИГЛАСАН МАТЕРИАЛ

1. Объект хандлагат технологийн С++ програмчлал, Ж.Пүрэв, 2008, Улаанбаатар.

# 7. ХАВСРАЛТ

#include <iostream>

#include <string.h>

using namespace std;

class employee

{                                    // employee class тодорхойлж байна

private:                             // private хандалтын түвшинтэй байна

    char \*name;                      // name хаяган хувьсагч зарлаж байна

    char \*alban\_tushaal;             // alban\_tushaal хаяган хувьсагч зарлаж байна

    float ajillasan\_t;               //ажилласан цаг float төрлийн утга авах хувьсагч зарлаж байна

    char \*ID;                        // ID хаяган хувьсагч зарлаж байна

public:                              // public хандалтын түвшин

    void set\_name(char \*s);          // name private хандалтын түвшинтэй хувьсагчид хандаж утга оноох функц

    void set\_alban\_tushaal(char \*s); // alban\_tushaal private хандалтын түвшинтэй хувьсагчид хандаж утга оноох функц

    char set\_ID(char \*s);            // ID private хандалтын түвшинтэй хувьсагчид хандаж утга оноох функц

    void copy(employee &s);          //хуулагч функц зарлаж байна

    void show\_data(void);            //мэдээллийг дэлгэцэлж харуулах функц

    char \*get\_name(void)

    {                // name private хандалтын түвшинтэй хувьсагчид хандах функц

        return name; //нэр буцаана

    }

    char \*get\_alban\_tushaal(void)

    {                         // alban\_tushaal private хандалтын түвшинтэй хувьсагчид хандах функц

        return alban\_tushaal; //албан тушаал буцаана

    }

    char \*get\_ID(void)

    {              // ID private хандалтын түвшинтэй хувьсагчид хандах функц

        return ID; // ID буцаана

    }

    employee();  //байгуулагч функц зарлаж байна

    ~employee(); //устгагч функц зарлаж байна

};

void employee::set\_name(char \*s)

{ // Set\_name функц тодорхойлолт

    if (name != NULL)

    { //санах ой нөөцөлсөн байсан эсэхийг шалгаад нөөцөлсөн байвал санах ойг чөлөөлнө

        delete[] name;

    }

    name = new char[strlen(s) + 1]; //шинэ санах ой нөөцлөнө. ингэхдээ орж ирсэн хүснэгтийн хэмжээгээр нөөцөлнө

    strcpy(name, s);                //нөөцөлсөн санах ой доо орж ирсэн утгыг хуулна

}

void employee::set\_alban\_tushaal(char \*s)

{ // Set\_alban\_tushaal функц тодорхойлолт

    if (alban\_tushaal != NULL)

    { //санах ой нөөцөлсөн байсан эсэхийг шалгаад нөөцөлсөн байвал санах ойг чөлөөлнө

        delete[] alban\_tushaal;

    }

    alban\_tushaal = new char[strlen(s) + 1]; //шинэ санах ой нөөцлөнө. ингэхдээ орж ирсэн хүснэгтийн хэмжээгээр нөөцөлнө

    strcpy(alban\_tushaal, s);                //нөөцөлсөн санах ой доо орж ирсэн утгыг хуулна

}

char employee::set\_ID(char \*s)

{ // Set\_ID функц тодорхойлолт

    if (ID != NULL)

    { //санах ой нөөцөлсөн байсан эсэхийг шалгаад нөөцөлсөн байвал санах ойг чөлөөлнө

        delete[] ID;

    }

    ID = new char[strlen(s) + 1]; //шинэ санах ой нөөцлөнө. ингэхдээ орж ирсэн хүснэгтийн хэмжээгээр нөөцөлнө

    strcpy(ID, s);                //нөөцөлсөн санах ой доо орж ирсэн утгыг хуулна

}

void employee::copy(employee &s)

{ // Set\_copy функц тодорхойлолт

    if(ID!=NULL){

        delete[] ID;

    }

    ID=new char[strlen(s.ID)+1];

    strcpy(ID,s.ID);

    if (name != NULL)

    { // name-д санах ой нөөцөлсөн байсан эсэхийг шалгаад нөөцөлсөн байвал санах ойг чөлөөлнө

        delete[] name;

    }

    name = new char[strlen(s.name) + 1]; //шинэ санах ой нөөцлөнө. ингэхдээ орж ирсэн хүснэгтийн хэмжээгээр нөөцөлнө

    strcpy(name, s.name);                //нөөцөлсөн санах ой доо орж ирсэн утгыг хуулна

    if (alban\_tushaal != NULL)

    { // alban\_tushaal-д санах ой нөөцөлсөн байсан эсэхийг шалгаад нөөцөлсөн байвал санах ойг чөлөөлнө

        delete[] alban\_tushaal;

    }

    alban\_tushaal = new char[strlen(s.alban\_tushaal) + 1]; //шинэ санах ой нөөцлөнө. ингэхдээ орж ирсэн хүснэгтийн хэмжээгээр нөөцөлнө

    strcpy(alban\_tushaal, s.alban\_tushaal);                //нөөцөлсөн санах ой доо орж ирсэн утгыг хуулна

}

employee::employee()

{                                //анхдагч байгуулагч функц тодорхойлолт

    ID = new char[5];            // ID-д 5 хэмжээтэй char төрлийн санах ой нөөцөлж байна

    strcpy(ID, "NULL");                 // ID-д NULL гэсэн утга оноож байна хамгийн сүүлийн байранд /0 байна

    name = new char[5];          // name-д 5 хэмжээтэй char төрлийн санах ой нөөцөлж байна

    strcpy(name, "NULL");               // name-д NULL гэсэн утга оноож байна хамгийн сүүлийн байранд /0 байна

    alban\_tushaal = new char[5]; // alban\_tushaal-д 5 хэмжээтэй char төрлийн санах ой нөөцөлж байна

    strcpy(alban\_tushaal, "NULL");     // alban\_tushaal-д NULL гэсэн утга оноож байна хамгийн сүүлийн байранд /0 байна

    ajillasan\_t = 0;             //Ажилласан цагт 0 утга оноох

}

employee::~employee()

{                           //устгагч функц тодорхойлолт

    delete[] name;          // name нөөцөлсөн санах ойг чөлөөлөх

    delete[] alban\_tushaal; // alban\_tushaal нөөцөлсөн санах ойг чөлөөлөх

}

void employee::show\_data()

{                                                         // show data функцийн тодорхойлолт хийж байна

    cout << " ID: " << get\_ID() << endl;                  //ажилласан цагийг хэвлэж харуулна

    cout << " name: " << get\_name() << endl;              //нэр массивт хадгалсан мэдээллийг хэвлэж харуулна

    cout << " position: " << get\_alban\_tushaal() << endl; //албан тушаал массивт хадгалсан өгөгдлийг хэвлэж харуулна

    // cout<<" tsalin: "<< tsalin;               //цалин хэвлэж харуулна

    cout << endl

         << endl; //мөр шилжүүлнэ

}

main()

{

    char a[20], b[20], c[20]; // 20хэмжээтэй хүснэгт үүсгэж байна

    employee d[5];            // 5 объект зарлаж байна

    for (int i = 0; i < 5; i++)

    { //давталт гүйлгэнэ. 5 объектэд утга оноохын тулд

    label1:

        int count = 0;

        cout << "number: " << i << endl;

        cout << "ID: ";

        cin >> a; //а хувьсагчид ID авна

        for (int j = 0; j <= i; j++)

        { // 0-i хүртэл гүйдэг нь өмнө нь давтагдсан эсэхийг шалгахын тулд

            if (strcmp(a, d[j].get\_ID()) == 0)

            { //энд давтагдсан эсэхийг шалгах үйлдэл хийж байна

                count = 1; //давтагдсан тохиолдол гарвал count нь 1 болно

                break;

            }

        }

        if (count == 0)

        {                   //давтагдсан тохиолдол гараагүй буюу count-ын анхны утга өөрчлөгдөөгүй байвал утга онооно

            d[i].set\_ID(a); // ID-д а хувьсагчид оруулсан утгыг хуулна.

            cout << "name: ";

            cin >> b;         // b массивт нэр хадгална

            d[i].set\_name(b); //нэрийг Set хийж объектэд дамжуулна

            cout << "position: ";

            cin >> c;                  // c массивт албан тушаал хадгална

            d[i].set\_alban\_tushaal(c); //хадгалсан утгыг албан тушаалд дамжуулна

        }

        else

        { // ID давтагдал гарсан тохиолдолд:

            cout << "-------------" << endl;

            cout << "burtgeltei hereglegc bn" << endl;

            cout << "-------------" << endl;

            goto label1; // label1 рү шилжих буюу дахин ID-д гараас утга авна.

        }

    }

    for (int i = 0; i < 5; i++)

    { //бүх мэдээллийг дэлгэцэнд харуулна

        d[i].show\_data();

    }

    for (int i = 0; i < 5; i++)

    { //эрэмбэлэх үйлдэл хийхийн тулд давхар давталт гүйж байна

        for (int j = 0; j < 4; j++)

        {

            if (strcmp(d[j].get\_name(), d[j + 1].get\_name()) > 0)

            {                   //нэрээр нь жишнэ

                employee tmp;   // tmp employee зарлаж байна. байрыг солиход хоосон хувьсагч хэрэг болдог

                tmp.copy(d[j]); //байрыг нь солихын тулд хуулагч функц ашиглан хуулж байна

                d[j].copy(d[j + 1]);

                d[j + 1].copy(tmp);

            }

        }

    }

    cout << "object exch" << endl;

    cout << "sorted" << endl;

    for (int i = 0; i < 5; i++)

    { //байр солигдсоны дараах мэдээллийг хэвлэж харуулна

        d[i].show\_data();

    }

    employee \*e[5]; //хаяган хүснэгт үүсгэж байна.

    for (int i = 0; i < 5; i++)

    {                        //

        e[i] = new employee; //

        e[i]->copy(d[i]);    //

    }

    for (int i = 0; i < 5; i++)

    { //эрэмбэлэх үйлдэл хийхийн тулд давхар давталт гүйж байна

        for (int j = 0; j < 4; j++)

        {

            if (strcmp(e[j]->get\_name(), e[j + 1]->get\_name()) > 0)

            {                  //нэрээр нь жишнэ

                employee \*tmp; // tmp employee зарлаж байна. байрыг солиход хоосон хувьсагч хэрэг болдог

                tmp = e[j];    //байрыг нь солихын тулд хуулагч функц ашиглан хуулж байна

                e[j] = e[j + 1];

                e[j + 1] = tmp;

            }

        }

    }

    cout << "pointer exch" << endl;

    cout << "sorted" << endl;

    for (int i = 0; i < 5; i++)

    { //байр солигдсоны дараах мэдээллийг хэвлэж харуулна

        e[i]->show\_data();

    }

}