# Private хандалтын түвшинтэй класс, объект хандлагат програмчлалын лабораторийн ажил (Лаборатори №4)

М.Солонго

ХШУИС, мэдээллийн системийн оюутан, 20b1num2104@stud.num.edu.mn

# 1. ОРШИЛ

Байгуулагч, анхдагч, устгагч функц, функц дахин тодорхойлох, хандалтын түвшин, динамик санах ойн талаар онолын мэдлэгийг жишээний хамт хавсарган өмнөх лабораторид тодорхойлсон классд анхдагч болон параметртэй байгуулагч нэмж тодорхойлж, захирлын цалин бодох функц, гишүүн өгөгдлүүдийг private хандалтын түвшинтэй болгож гишүүн өгөгдөлд хандах set болон get функц бичин эцсийн үр дүнг дүгнэв.

# 2. ЗОРИЛГО

Объектын классыг шаардлагын дагуу тодорхойлж гишүүн өгөгдөл болон захирлын цалин бодох функцийг private хандалтын түвшинтэй тодорхойлох, түүнд хандах функц тодорхойлохын тулд дараах зорилтуудыг тавьж ажилласан:

1. Private, public, protected хандалтын түвшингийн ялгааг таниж мэдэн хэрэглэх
2. Гишүүн өгөгдлүүдийн төрлийг тодорхойлох
3. Параметртэй байгуулагч функц тодорхойлох
4. Устгагч функц тодорхойлох үед хэзээ дуудагдаж байгааг туршилтад үндэслэн хариулах

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 Байгуулагч функц

Объект байгуулах үед гишүүн өгөгдөлд санах ой хадгалж гарааны утга оноох үйлдэл хийдэг функц. Классын нэртэй ижил байна. Уг функц ямар ч утга буцаадаггүй. Void гэж зарлах шаардлагагүй. Учир нь төрөл нь анхнаасаа void төрлөөр өгөгддөг. Тусгай зориулалттай функц бөгөөд public хандалтын түвшинтэй байна. Дахин програмчилж болдог.

## 3.2 Анхдагч байгуулагч

Ямар нэгэн байгуулагч функц тодорхойлж өгөөгүй байвал c++ compiler өөрөө байгуулагч функц нэмэлтээр тодорхойлж өгдөг. Үүнийг анхдагч байгуулагч гэнэ.

## 3.3 Устгагч функц

Объект санах ойгоос чөлөөлөгдөх үед нөөцөлсөн санах ойг чөлөөлөх үүрэгтэй функцийг хэлнэ. Байгуулагч функцтай адил классын нэртэй ижил байна. Гэхдээ урдаа ~ тэмдэг тавина. Устгагч функцийг дахин програмчилж болохгүй. Тодорхойлж өгөөгөүй тохиолдолд

## 3.3 Функц дахин тодорхойлох

Функцийн нэр адил байж болно. Харин параметрийн тоо, төрөл, дараалал өөр байж болно. Үүнийг дахин програмчилал гэнэ. Жишээлбэл:

Void get\_data(int x){

}

Void get\_data(float x,float y){

}

## 3.4 Хандалтын түвшин

Хандалтын 3 түвшин байна.

1. Private:

Private түвшинээр тодорхойлсон гишүүн өгөгдөл, гишүүн функцэд гаднаас хандаж болохгүй.

1. Protected

Удамшдаг. Private хандалтын түвшинтэй адилаар гаднаас хандахыг хориглодог.

1. Public

Классын гишүүн өгөгдөлд гадна талаас хандахыг зөвшөөрдөг.

## Динамик санах ой

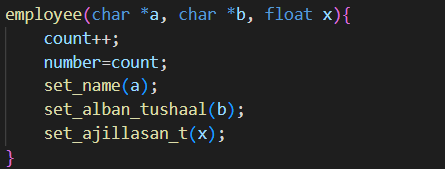
Char a[20] хэмжээтэй массив зарласан мөртлөө 20-оос дээш эсвэл доош хэмжээний бичвэрэн мэдээлэл оруулвал overflow, underflow үүсдэг. Үүнээс сэргийлэхийн тулд new оператор ашиглаж оруулсан өгөгдлийн хэмжээгээр санах ой үүсгэх боломжтой.

# 4. ХЭРЭГЖҮҮЛЭЛТ

## 1. Анхдагч байгуулагч

## 

Параметртэй байгуулагч

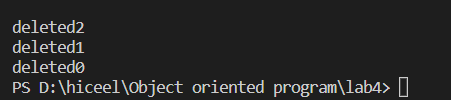


Устгагч функц

## 

Байгуулагч функц бичин үүсгэгдэх бүрт count нэмэгдэх байдлаар тодорхойлсон ба устгагч функц хэзээ хэрхэн ажиллаж байгааг харахын тулд count хасагдаж байгаагаар хэвлэн үзлээ.





Үүнээс код ажиллаж эхлэхэд анхдагч байгуулагч үүсгэгдэж, устгагч функц санах ойг хамгийн сүүлд чөлөөлж байгааг харж байна.

# 5. ДҮГНЭЛТ

Байгуулагч функц нь main функц дээр олон объект зарлалт хийж утга оноохыг хялбарчилж өгдөгөөс гадна шинээр санах ой үүсгэсэн бол чөлөөлөхгүй шууд хаягийг нь хадгалж байсан хувьсагчид өөр утга оноож өгөх юм бол ойн цоорхой үүснэ гэдгийг маш сайтар ойлгож авлаа.

# 6. АШИГЛАСАН МАТЕРИАЛ

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

# 7. ХАВСРАЛТ

#include<iostream>

#include<string.h>

using namespace std;

int count=0;                                            //static хувьсагч зарлаж байна. энэ нь нийт хэдэн удаа байгуулагч функц үүсгэгдэж байгааг тоолохоос гадна ажилчдыг дугаарлах үүрэгтэй

class employee{                                         //employee класс зарлаж байна

    private:                                            //private хандалтын түвшинтэй өгөгдөл функцуудыг зарлана

        char \*name;                                     //нэр хадгалах хүснэгтэн өгөгдөл зарлаж байна.

        char \*alban\_tushaal;                            //албан тушаал гэсэн бичвэрэн өгөгдөл хадгалах массив зарлаж байна

        float time;                                     //ажилласан цаг хувьсагч буюу гишүүн өгөгдөл зарлаж байна

        float zahirlin\_tsalin\_bodoh(void){              //захирлын цалин бодох функц

            return tsalin+=50000;                       //Цалин дээр захирлын цалингийн нэмэгдэлийг нэмж буцаана

        }

    public:                                             //public хандалтын түвшинтэй өгөгдөл функцуудыг зарлана

        int number;                                     //number нэртэй гишүүн өгөгдөл зарлаж байна

        float tsalin;                                   //цалин нэртэй гишүүн өгөгдөл

        void show\_data(void);                           //мэдээлэл дэлгэцэнд харуулах функц зарлаж байна

        float tsalin\_bodoh(void);                       //цалин бодох үйлдэл хийх функц зарлалт

        float ajillasan\_tsag\_nemegduuleh(float x);      //ажилласан цагийг нэмэгдүүлэх үйлдэл хийнэ

        void copy(employee &s);                         //copy функц зарлалт. энэ нь ойн цоорхой үүсгэхээс сэргийлнэ

        void set\_name(char \*s){                         //name нэртэй private хандалтын түвшинтэй гишүүн өгөгдөлд утга оноох функц

            if(name != NULL){                           //нэр хаяган хувьсагч хоосон эсэхийг шалгаж байна

                delete[] name;                          //хоосон биш байгаа тул Delete operator ашиглан санах ойг чөлөөлнө

            }

            name = new char[strlen(s)+1];               //name-д шинэ санах ой нөөцлөнө

            strcpy(name, s);                            //нөөцөлсөн санах ойд S хаяган хувьсагчийн зааж байгаа хүснэгтийг оруулна

        }

        char \*get\_name(void){                           //name нэртэй private хандалтын түвшинтэй гишүүн өгөгдөлд хандах функц

            return name;                                //утга буцаана

        }

        void set\_alban\_tushaal(char \*s){                //албан тушаал нэртэй private хандалтын түвшинтэй гишүүн өгөгдөлд утга оноох функц

            if(alban\_tushaal != NULL){                  //албан тушаал хаяган хувьсагчинд санах ой нөөцөлсөн байна уу шалгана

                delete[] alban\_tushaal;                 //санах ой нөөцөлсөн байвал устгаж байгаад шинийг нөөцөлж болно. чөлөөлөөгүй тохиолдолд ойн цоорхой үүснэ

            }

            alban\_tushaal = new char[strlen(s)+1];      //шинэ санах ой нөөцлөнө

            strcpy(alban\_tushaal, s);                   //албан тушаалд S хаяган хувьсагчийн зааж байгаа char хүснэгтийг оруулна

        }

        char \*get\_alban\_tushaal(void){                  //албан тушаал нэртэй private хандалтын түвшинтэй гишүүн өгөгдөлд хандах функц

            return alban\_tushaal;                       //албан тушаал буцаана

        }

        void set\_ajillasan\_t(float x){                  //ажилласан цаг нэртэй private хандалтын түвшинтэй гишүүн өгөгдөлд утга оноох функц

                time=x;                                 //ажилласан цаг гишүүн өгөгдөлд утга онооно

        }

        float get\_ajillasan\_t(void){                    //ажилласан цаг нэртэй private хандалтын түвшинтэй гишүүн өгөгдөлд хандах функц

            return time;                                //ажилласан цаг буцаана.

        }

        employee(){                                     //employee анхдагч байгуулагч функц тодорхойлж байна

            cout << "created: " << count++ <<"\n";      //хэдэн байгуулагч функц үүсгэгдэж байгааг тоолох тоолуур

            number=count;                               //дугаарт тоолуур оноож өгч байна. ингэснээр шууд дугаарлагдана

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

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

            name = "NULL";                              //5 хэмжээтэй санах ой нөөцөлсөн name-д NULL ийг оруулж өгнө

            alban\_tushaal = "NULL";                     //5 хэмжээтэй санах ой нөөцөлсөн alban\_tushaal-д NULL ийг оруулж өгнө

            time = 0;

        }

        employee(char \*a, char \*b, float x){            //параметртэй байгуулагч функц тодорхойлж байна

            count++;                                    //байгуулах бүрд cout тоолуурын гишүүн нэмэгдэнэ

            number=count;                               //дугаарт тоолуур оноож өгч байна. ингэснээр шууд дугаарлагдана

            set\_name(a);                                //name -д а хүснэгтэд байгаа үгийг хадгална

            set\_alban\_tushaal(b);                       //албан тушаалд b хүснэгтэд байгаа үгийг хадгална

            set\_ajillasan\_t(x);                         //ажилласан цаг тохируулах функцийг дуудаж private хандалтын түвшинтэй гишүүн өгөгдөлд утга онооно

        }

        ~employee(){                                    //устгагч функц тодорхойлж байна

            cout << "deleted"<< --count <<endl;         //нийт хэдэн удаа байгуулагч функц устгагдсаныг харуулна

            delete[] name;                              //name-д нөөцөлсөн санах ойг чөлөөлөх үүрэгтэй

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

        }

};

void employee::show\_data(){                             //show data функцийн тодорхойлолт хийж байна

    cout<<" dugaar: "<<number<<endl;                    //дугаарт хадгалсан мэдээллийг хэвлэж харуулна

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

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

    cout<<" ajillasan tsag: "<<get\_ajillasan\_t()<<endl; //ажилласан цагийг хэвлэж харуулна

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

    cout<<endl<<endl;                                   //мөр шилжүүлнэ

}

float employee::tsalin\_bodoh(){                         //цалин бодох функцийн тодорхойлолт

    int ts=6500;                                        //цалинг цагийн 6500 аар тооцон бодохын тулд ts зарлаж өгсөн

    if(strcmp(alban\_tushaal, "zahiral") == 0){          //албан тушаал нь захирал гэсэн үгтэй адилхан байна уу гэдгийг шалгана

        tsalin =time\*ts;                                //цалинг бодохдоо ажилласан цагийг тогмолоор үржүүлж гаргана

        return zahirlin\_tsalin\_bodoh();                 //захирлын цалин бодох функцийг дуудаж ажиллуулна

    }else{

        tsalin =time\*ts;                                //эсрэг тохиолдолд ажилчдын цалин бодно. ажилласан цагийг тогтмолоор шууд үржүүлээд ямар нэгэн нэмэгдэл байхгүй

        return tsalin;                                  //цалинг буцаана

    }

}

float employee::ajillasan\_tsag\_nemegduuleh(float x){    //ажилласан цаг нэмэгдүүлэх функц тодорхойлж байна

    if(x<24 && x>0){                                    //параметрээс орж ирж буй утга 0ээс24 ын хоорон байна уу шалгах

        time += x;                                      //нөхцөл биелэсэн тохиолдолд ажилласан цагийг нэмэгдүүлэх үйлдэл хийнэ

        return time;                                    //ажилласан цагийг буцаана

    }else{

        cout<<"error"<<endl;                            //эсрэг тохиолдолд Error гэсэн утга хэвлэж харуулж ямар нэгэн нэмэгдэл орохгүй

        return time;                                    //time утга буцаана

    }

}

void employee::copy(employee &s){                       //copy функц тодорхойлж байна

    if(name!=NULL){                                     //name-д char төрөлтэй мэдээлэл байна уу шалгаад

        delete []name;                                  //байвал санах ойг чөлөөлнө

    }

    name=new char[strlen(s.name)+1];                    //name-д шинэ санах ой нөөцөлж байна

    strcpy(name,s.name);                                //хадгалсан санах ойн доо employee s-ийн нэрийг хуулна

    if(alban\_tushaal!=NULL){                            //албан тушаал-д char төрөлтэй мэдээлэл байна уу шалгаад

        delete []alban\_tushaal;                         //байвал санах ойг чөлөөлнө

    }

    alban\_tushaal=new char[strlen(s.alban\_tushaal)+1];  //албан тушаал-д шинэ санах ой нөөцөлж байна

    strcpy(alban\_tushaal,s.alban\_tushaal);              //хадгалсан санах ойн доо employee s-ийн албан тушаалыг хуулна

    time=s.time;                                        //employee s-ийн ажилласан цагийг хуулна

    tsalin=s.tsalin;                                    //цалинг хуулна

}

main(){

    int i;                                              //i тоолууран хувьсагч зарлаж байна

    float o;                                            //3 ажилчдын хүснэгт үүсгэж байна. анхдагч байгуулагчийн тусламжтайгаар

    employee c[3];

    for(i=0;i<3;i++){                                   //давталт гүйлгэнэ. 0ээс 3 хүртэл

        char name[20],ATu[20];                          //давталт гүйх болгонд name ATu массив үүсгэгдэнэ

        float k;                                        //ажилласан цагийг гараас авах хувьсагч зарлалт

        cout<<c[i].number<<endl;                        //ажилчны дугаарыг хэвлэж харуулна

        cout<<"name: ";                                 //name: гэж хэвлэж харуулна

        cin>>name;                                      //name массивт гараас утга авна

        c[i].set\_name(name);                            //set\_name тусламжтайгаар name-д гараас авсан хүснэгтийн хэмжээтэй санах ой үүсгэгдэж хадгалагдана

        cout<<"alban tushaal: ";                        //alban\_tushaal: гэж хэвлэж харуулна

        cin>>ATu;                                       //ATu массивт гараас утга авна

        c[i].set\_alban\_tushaal(ATu);                    //set\_alban\_tushaal тусламжтайгаар alban\_tushaal-д гараас авсан хүснэгтийн хэмжээтэй санах ой үүсгэгдэж хадгалагдана

        cout<<"ajillasan tsag: ";                       //ajillasan tsag: гэж хэвлэж харуулна

        cin>>k;                                         //k хувьсагчид гараас утга авна

        c[i].set\_ajillasan\_t(k);                        //set\_ajillasan\_t функцийн тусламжтайгаар ажилласан цаг private хандалтын түвшинтэй гишүүн өгөгдөлд утга оруулна

    }

    for(i=0;i<3;i++){                                   //For давталт 0-3 хүртэл гүйнэ

        c[i].show\_data();                               //бүх ажилчдын мэдээллийг хэвлэж харуулна

    }

    // cout<<"ajillasan tsag nemegduuleh"<<endl;

    // cout<<"ajilcnii dugaar: ";

    // cin>>d;

    // cout<<"ajillasan tsag: ";

    // cin>>o;

    // c[d].ajillasan\_tsag\_nemegduuleh(o);

    for(i=0;i<3;i++){                                   //давхар давталт гүйнэ

        for(int j=0;j<2;j++){

            if(c[j].tsalin>c[j+1].tsalin){              //0 индекс дэх ажилчин 1 индекс дэх ажилчны цалингаас их байна уу гэдгийг шалгах

                employee tmp;                           //нөхцөл биелэж байвал байрыг copy функцийн тусламжтайгаар солих үйлдэл хийнэ

                tmp.copy(c[j]);

                c[j].copy(c[j+1]);

                c[j+1].copy(tmp);

            }

        }

    }

    for(i=0;i<3;i++){                                   //ажилчдын мэдээллийг хэвлэхийн тулд for давталт гүйлгэж байна

        c[i].show\_data();                               //хэвлэж харуулах функц дуудаж байна.

    }

}