

КЛАСС, ОБЪЕКТ БА ТҮҮНИЙГ ТОДОРХОЙЛОХ (ЛАБОРАТОРИ №3)

Э.Уранчимэг

ХШИУС, МКУТ, Программ хангамжийн III түвшний оюутан,
21B1NUM0609@stud.num.edu.mn

1. ОРШИЛ

Энэ тайланд класс, түүний онцлог, үүрэг болон C++ объектын ялгааны талаар дэлгэрэнгүй тайлбарласан. Мөн гишүүн функц, гишүүн өгөгдөлийн хамаарал, түүнд хэрхэн хандах болон өгөгдлийн битүүмжлэлийн талаар судлан тайлбарласан. Хэрэгжүүлэлт хэсэгт класс тодорхойлон харуулсан. Хавсралт хэсгээс дэлгэрэнгүй кодыг унших боломжтой.

2. ЗОРИЛГО

Классыг ашиглан объектуудад хандаж, түүнийг тодорхойлж, мөн классыг тодорхойлох, дуудан хэрэгжүүлсэн. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Классыг тодорхойлох,
2. C++ -ийн объектыг тодорхойлох,
3. Классын гишүүн өгөгдлийг тодорхойлох,
4. Классын гишүүн функцийг тодорхойлох,
5. Класс ашиглах программ бичих

3. ОНОЛЫН СУДАЛГАА

3.1 Класс гэж юу болох, онцлог, үүрэг

Класс гэдэг нь объектыг хийсвэрлэх зорилго бүхий үүсмэл өгөгдлийн төрөл юм. Үүсмэл өгөгдлийн төрлийг программ бичиж буй хүмүүс өөрсдөө зохиодог. Тооноос өөр зүйл төлөөлүүлэхийн тулд ашигладаг.

Класс нь програм хангамжийн системд хэд хэдэн чухал үүрэг, хариуцлагыг гүйцэтгэдэг:

- Классын үндсэн үүрэг бол түүнээс үүсгэж болох объектуудын бүтэц, үйл ажиллагааг тодорхойлох явдал юм. Энэ нь объектуудад байх шинж чанарууд (өгөгдөл) болон аргууд (зан үйл) -ийг тодорхойлдог. Энэ нь кодын тогтвортой байдал, дахин ашиглах боломжтой байдлыг дэмждэг.
- Классууд нь өгөгдөл, аргуудыг багтаасан бөгөөд энэ нь өгөгдлийг хэрхэн хадгалах, удирдах дотоод нарийн ширийнийг нууцалдаг. Энэ нь өгөгдлийн

бүрэн бүтэн байдлыг хадгалах, сайн тодорхойлсон интерфейсээр дамжуулан өгөгдөлд хандах хандалтыг хянахад тусалдаг.

- Классууд нь удамшлын шатлалд оролцдог ба эндээс та эцэг ангиудаас шинж чанарууд болон аргуудыг өвлөн авах дэд ангиудыг үүсгэж болно. Энэ нь кодын дахин хэрэглээг дэмжиж, объект хоорондын харилцааг загварчлах боломжтой болгоно.
- Классууд нь полиморфизмыг идэвхжүүлдэг бөгөөд полиморфизм нь нийтлэг дээд ангитай бол өөр өөр ангиллын объектуудыг зэрэг дуудах боломжтой болгодог. Энэ нь таны кодын уян хатан байдлыг дэмждэг.
- Сайн тодорхойлсон ангийн тодорхойлолтууд нь тухайн ангийн объектуудыг хэрхэн ашиглах талаар баримт бичиг болдог. Энэ нь бусад хөгжүүлэгчид (эсвэл өөрөө ч) таны кодыг ойлгож ашиглахад хялбар болгодог.

3.2 Класс болон C++ объектын ялгаа

Класс:

- ★ Объектыг үүсгэх загвар болж өгдөг.
- ★ Классыг байгуулахад санах ой шаардаггүй
- ★ Тухайн классыг нэг л удаа үүсгэдэг

Объект:

- ★ Классын загвараар үүсдэг.
- ★ Санах ой шаарддаг.
- ★ Тухайн классын загварыг ашиглан хэдэн ч объект үүсгэж болдог.

3.3 Гишүүн функц, гишүүн өгөгдөл хоёр ямар хамааралтай байдаг вэ?

Гишүүн өгөгдөл гэдэг нь объектын шинж чанар, гишүүн функц гэдэг нь объектын үйл хөдлөл юм. 1 үйлчлэх хүрээн дотор хамаарч байгаа гишүүн функц дотроос гишүүн өгөгдөл рүү шууд хандаж болно.

3.4 Өгөгдлийн битүүмжлэл гэж юу вэ?

Классын гишүүн өгөгдөл нь ихэвчлэн Private түвшинд тодорхойлогддог. Private нь зөвхөн тухайн класс дотроос уг өгөгдөлд хандаж болох шинжийг илэрхийлдэг. Иймд классын гишүүн функцуудаар дамжуулан түүний гишүүн өгөгдөл дээр үйлдэл хийж болдог. Өгөгдлийн битүүмжлэл гэдэг нь өөр классын функцуудаас тухайн классын өгөгдлөд хандахаас сэргийлэхийг хэлэх ба зөвхөн уг классын функцууд нь гишүүн өгөгдөлд хандаж болох горим. Энэ нь санамсаргүй байдлаар классын гишүүн өгөгдөлд өөрчлөлт гарахаас сэргийлж байгаа.

3.5 Классын гишүүн өгөгдөл болон гишүүн функцэд хэрхэн хандах вэ?

Гишүүн өгөгдөлд хандахдаа объектын нэрийг бичээд шууд хандалтын операторыг бичээд өгөгдлийн нэрийг бичнэ. Гишүүн функцд хандахдаа мөн адил объектын нэр, шууд хандалтын оператор, функцын нэр параметрыг бичнэ. Функцын параметрт заалт эсвэл заагчаар утга дамжуулах нь ашигтай.

4. ХЭРЭГЖҮҮЛЭЛТ

1. Ажилчин гэсэн класс тодорхойлно.

```
class worker{
```

Энэ класс нь хувийн дугаар, нэр, албан тушаал, ажилласан цаг гэсэн гишүүн өгөгдөлтэй ба тохирох өгөгдлийн төрлөөр нь private хандалтын түвшинтэйгээр тодорхойлж өгсөн. Нэр нь 20 тэмдэгт байх ба албан тушаал нь 10 тэмдэгт байхаар тодорхойлсон.

```
private:
    int number;
    char name[20];
    char pos[10];
    float time;
```

Мөн уг класст гарааны утга оноох(), гараас утга авах(), мэдээлэл дэлгэцлэх(), цалин бодох(), захирлын цалин бодох(), ажилласан цаг нэмэгдүүлэх() гэсэн гишүүн функцүүд байх ба эдгээрийг public хандалтын түвшинтэйгээр зааврын дагуу класс дотроо зарлан, тодорхойлж өгсөн.

```
public:
    void setInfo() {
        number=0;
        strcpy(name, "");
        strcpy(pos, "ajilchin");
        time=0;
    }

    void changeInfo() {
        cout<<"ID:";
        cin>>this->number;
        cout<<"Ner:";
        cin>>this->name;
        cout<<"Alban tushaal:";
        cin>>this->pos;
```

```

        cout<<"Ajillasan tsag:";
        cin>>this->time;
    }

    void showInfo() {
        cout<<"ID:"<<this->number<<endl;
        cout<<"Ner:"<<this->name<<endl;
        cout<<"Alban tushaal:"<<this->pos<<endl;
        cout<<"Ajillasan tsag:"<<this->time<<endl;
    }

    float zahiralSalary() {
        return this->time * 5000;
    }

    float salary() {
        float sal = this->time * 3500;
        if(strcmp(this->pos, "zahiral") == 0) {
            sal += this->zahiralSalary();
        }
        return sal;
    }

    bool addTime(float time) {
        if(time >=0 && time <= 24) {
            this->time += time;
            return 1;
        }
        return 0;
    }
}

```

2. Ажилчдын ажилласан цаг бүрийг өөрчилж цалин тооцоолох

Гол функцдээ ажилчдынхаа тоог авч worker гэсэн классын төрөлтэй массив зарласан ба энэ массивд worker төрлийн өгөгдлүүдийг хадгална.

```

int main() {
    int n;
    cout << "Heden ajiltantai ve?" << endl;
    cin >> n;
    worker a[n];
}

```

Давталтаар ажилчдынхаа мэдээллийг гараас утга оруулан авч, мөн мэдээллийг дэлгэцлэх ба ажилчдын цагийг өөрөөр нэмэгдүүлж, ажилчдын мэдээллийг ажин дэлгэцлэх программыг бичсэн.

```
for (i = 0; i < n; i++) {
    cout << i + 1 << "-r Ajilchinii medeelel avah:" << endl;
    a[i].changeInfo();
}

for (i = 0; i < n; i++) {
    cout << i + 1 << "-r Ajilchinii medeelel:" << endl;
    a[i].showInfo();
    cout <<a[i].salary()<<endl;
}

for (i = 0; i < n; i++) {
    cout << i + 1 << "-r Ajilchinii tsag nemegduuleh" << endl;
    float addTime;
    cin >> addTime;
    a[i].addTime(addTime);
    cout << endl;
}

for (i = 0; i < n; i++) {
    cout << i + 1 << "-r Ajilchinii medeelel:" << endl;
    a[i].showInfo();
    cout <<a[i].salary()<<endl;
    cout << endl;
}

return 0;
```

Программ ажилуулсан туршилтын үр дүн:

Heden ajiltantai ve?

4

1-r Ajilchinii medeelel avah:

ID:203

Ner:Enhee

Alban tushaal:Nygtlan

Ajillasan tsag:200

2-r Ajilchinii medeelel avah:

ID:204

Ner:Urana

Alban tushaal:Zahirai

Ajillasan tsag:150

3-r Ajilchinii medeelel avah:

ID:201

Ner:Oyuka

Alban tushaal:Ajlchin

Ajillasan tsag:175

4-r Ajilchinii medeelel avah:

ID:205

Ner:Urlug

Alban tushaal:Zahirai

Ajillasan tsag:200

1-r Ajilchinii medeelel:

ID:203

Ner:Enhee

Alban tushaal:Nygtlan

Ajillasan tsag:200

700000

2-r Ajilchinii medeelel:

ID:204

Ner:Urana

Alban tushaal:Zahirai

Ajillasan tsag:150

525000

3-r Ajilchinii medeelel:

ID:201

Ner:Oyuka

Alban tushaal:Ajlchin

Ajillasan tsag:175

612500

4-r Ajilchinii medeelel:

ID:205

Ner:Urlug

Alban tushaal:Zahirai

Ajillasan tsag:200

700000

1-r Ajilchinii tsag nemegduuleh

1.5

2-r Ajilchinii tsag nemegduuleh

1.3

3-р Ажилчиний тсэг нэмэгдүүлэх
1.6

4-р Ажилчиний тсэг нэмэгдүүлэх
1.2

1-р Ажилчиний мэдээлэл:
ID:203
Ner:Enhee
Alban tushaal:Nygtlan
Ajillasan tsag:201.5
705250

2-р Ажилчиний мэдээлэл:
ID:204
Ner:Urana
Alban tushaal:Zahirai
Ajillasan tsag:151.3
529550

3-р Ажилчиний мэдээлэл:
ID:201
Ner:Oyuka
Alban tushaal:Ajilchin
Ajillasan tsag:176.6
618100

4-р Ажилчиний мэдээлэл:
ID:205
Ner:Urlug
Alban tushaal:Zahirai
Ajillasan tsag:201.2
704200

5. ДҮГНЭЛТ

Класс тодорхойлсноор гарч болох давуу тал болон түүний ялгааг мэдсэнээр бага үйлдлээр их зүйлийг хийх боломжтой болохоос гадна программд бичиг баримт болох, өгөгдөлд хандах хандалтыг хянах зэрэгт илүү тустай.

6. АШИГЛАСАН МАТЕРИАЛ

- “Объект хандлагат программчлал лекцийн материал”, М.Золжаргал
- <https://www.techtarget.com/whatis/definition/class#:~:text=In%20object%2Doriented%20programming%2C%20a,ideas%20of%20object%2Doriented%20programming.>

7. ХАВСРАЛТ

```
#include <iostream>
#include <cstring>
using namespace std;

class worker{
private:
    int number;
    char name[20];
    char pos[10];
    float time;
public:
    void setInfo() {
        number=0;
        strcpy(name, "");
        strcpy(pos, "ajilchin");
        time=0;
    }

    void changeInfo() {
        cout<<"ID:";
        cin>>this->number;
        cout<<"Ner:";
        cin>>this->name;
        cout<<"Alban tushaal:";
        cin>>this->pos;
        cout<<"Ajillasan tsag:";
        cin>>this->time;
    }

    void showInfo() {
```



```

        cout<<"ID:"<<this->number<<endl;
        cout<<"Ner:"<<this->name<<endl;
        cout<<"Alban tushaal:"<<this->pos<<endl;
        cout<<"Ajillasan tsag:"<<this->time<<endl;
    }

    float zahiralSalary() {
        return this->time * 5000;
    }

    float salary() {
        float sal = this->time * 3500;
        if(strcmp(this->pos, "zahiral") == 0) {
            sal += this->zahiralSalary();
        }
        return sal;
    }

    bool addTime(float time) {
        if(time >=0 && time <= 24) {
            this->time += time;
            return 1;
        }
        return 0;
    }
};

int main() {
    int n;
    cout << "Heden ajiltantai ve?" << endl;
    cin >> n;
    int i;
    worker a[n];
    for (i = 0; i < n; i++) {
        cout << i + 1 << "-r Ajilchinii medeelel avah:" << endl;
        a[i].changeInfo();
    }

    for (i = 0; i < n; i++) {
        cout << i + 1 << "-r Ajilchinii medeelel:" << endl;

```

```
        a[i].showInfo();
        cout <<a[i].salary()<<endl;
    }
    for (i = 0; i < n; i++) {
        cout << i + 1 << "-r Ajilchinii tsag nemegduuleh" << endl;
        float addTime;
        cin >> addTime;
        a[i].addTime(addTime);
        cout << endl;
    }
    for (i = 0; i < n; i++) {
        cout << i + 1 << "-r Ajilchinii medeelel:" << endl;
        a[i].showInfo();
        cout <<a[i].salary()<<endl;
        cout << endl;
    }
    return 0;
}
```