

(Лаборатори №3)

Э.Багабанди

ХШУИС - ийн 3-р түвшиний оюутан, 19B1NUM0700@num.edu.mn

1. ОРШИЛ/УДИРТГАЛ

Уг лабораторийн хүрээнд өгөгдсөн асуултуудад хариулж, тайлбарлахад ашиглагдсан C++ хэлний онолын ойлголт(класс, объект, тэдгээрийн ялгаа болон хэрхэн тодорхойлох, гишүүн функц, гишүүн өгөгдөл, өгөгдлийн битүүмжлэл, функцийн параметрт гарааны утга оноох, байгуулагч функц тодорхойлох) - ыг судалж, үндсэн ойлголтуудыг эзэмшсэн болно.

2. ЗОРИЛГО

Энэхүү лабораторийн ажлын хүрээнд C++ хэлний класс, объект, тэдгээрийн хэрэглээ болон ялгааг ойлгох, гишүүн функц тодорхойлох, гишүүн өгөгдөл өгөгдлийн битүүмжлэл, байгуулагч функц тодорхойлох, функцийн параметрт гарааны утга оноох онолын ойлголтуудыг судалж, үүнийг ашиглан лабораторт өгөгдсөн бодлого болон асуултуудыг шийдвэрлэж, гарч буй үр дүн болон онолуудыг тайлбарлана.

3. ОНОЛЫН СУДАЛГАА

3.1 Класс ба объект

▪ Объект

- Объект нь өөрийн гэсэн шинж тэмдэг, үйл хөдлөлтэй байх ба объектын шинжийг өгөгдөл, үйл хөдлөлийг нь функц гэж нэрлэдэг. Объект нь ямар нэгэн зүйл хийнэ. Тэр хөдөлж, өөрийгөө хүрээлэн байгаа орчны объектуудад нөлөөлж чадна. Иймд объект нь өгөгдөл ба өгөгдөл боловсруулалт хийх функцтэй байх бөгөөд уг функцийг объектийн гишүүн функц, харин өгөгдлийн объектийн гишүүн өгөгдөл гэнэ.

▪ Класс

- Класс нь объектийн загвар бөгөөд объектийг уг загварын дагуу бүтээнэ. Класс нь гишүүн өгөгдөл, гишүүн нэгэн цогц нэгдэл юм.

▪ Зарлах, тодорхойлох

- Класс зарлах: class нь түүний дараах үгийг классын нэр болохыг заах үүрэгтэй тусгай үг юм. Классын ишүүдийн тодорхойлолт жагсаалтын хэсэг нь нээх их хаалтаар эхэлж, ардаа цэг таслал бүхий хаах их хаалтаар төгсөнө. Классын гишүүдийн тодорхойлолт нь private/хувийн/, public/нийтийн/, protected/хязгаарлагдмал/ хэмээх гурван хэсэгтэй байж болно.

- Класс тодорхойлох: Гишүүн өгөгдлийг тодорхойлж, функцийг гол төлөв зарлана. Дараах загварт үзүүлснээр тодорхойлж болно. Классын тодорхойлолтоор ой бэлддэггүй. Энэ бол зөвхөн классын хувьсагч буюу объект үүсгэх логик загвар юм.

```
class class_name
{
    private/protected/public:
        data_members;

    private/public:
        members_functions;
};
```

- Классын гишүүн функц тодорхойлох: Классын гишүүн функцийг тодорхойлолтыг ихэвчлэн классын тодорхойлолтын гадна талд хийнэ. Мөн гишүүн функцийг классын тодорхойлж дотор тодорхойлох боломжтой. Дараах үзүүлсэн загвараар тодорхойлж болно.

```
type class_name::function_name (parameters) {...}
```

- Объект: main() функц дотор **class_name a;** гэж бичсэнээр тухайн классын объект үүсэх юм.

3.2 Гишүүн функц ба гишүүн өгөгдөл, өгөгдөл битүүмжлэл

- **Гишүүн өгөгдөл**
 - Бүтэц гишүүн өгөгдөлтэй байдаг шиг класс бас гишүүн өгөгдөлтэй байна.
- **Гишүүн функц**
 - Гишүүн функц классынхаа гишүүн өгөгдөл дээр боловсруулалт хийж програмын бусад функцтэй мэдээлэл солилцоно.
- **Хэрхэн хандах**
 - Гишүүн өгөгдөлд хандах: хэрэв өгөгдөл public бол шууд **object_name.data** гэж хандах боломжтой. Харин private тохиолдолд шууд хандах боломжгүй юм.
 - Гишүүн функцид хандах: хэрэв функц public бол шууд **object_name.function_name()** гэж хандах боломжтой. Харин private тохиолдолд шууд хандах боломжгүй юм.
- **Өгөгдлийн битүүмжлэл**
 - Объектын утга далд байх бөгөөд энэ нь түүнийг бусдаас нуух гэсэн санаа биш. Харин объектын өгөгдөлд гишүүн функцээр нь дамжуулж хандахыг хэлнэ. Объектын өгөгдөлд тухайн классын биш функцээс хандах боломжгүй бөгөөд энэ нь гишүүн өгөгдөл

санаандгүй үйлдлээс хамгаалагдана. Өгөгдөл, түүн дээр үйлдэл хийх функцийг хамтад нь холбоотой авч үзэх ОХП – ын нэгэн чухал шинж юм. Өгөгдөл ба функц нь класс гэгдэх нэгэн цогц ойлголтонд нэгддэг. Иймд **битүүмжлэл** бол функц, өгөгдлийг тодорхой түвшинд нээлттэй биш объектийн хүрээнд хамтад нь холбоотойгоор авч үздэг ойлголт юм.

3.3 Функцийн параметрт гарааны утга оноох

- Функцийг зарлах болон тодорхойлохдоо түүний параметруудийн гарааны утгуудыг дараах байдлаар зааж өгч болно. Жишээ нь: `int myfunction (int a = 10, int b = 20, int c = 30, int d = 40);`
- Гарааны утгатай параметруудийг жагсаалтын сүүлд бичиж өгөх ёстойв. Аль нэгэн параметрийн утгыг хэрэглэх бол түүний дараах бүх параметруудийн гарааны утгыг бас хэрэглэх ёстой.

3.4 Байгуулагч ба устгагч функц

- **Байгуулагч функц**
 - Байгуулагч функц нь класстайгаа ижил нэртэй байна. Гишүүн өгөгдөлд ой бэлдэх, гарааны утга олгох зорилготойгоор уг функцийг тодорхойлдог. Объектийг олон янзаар байгуулах боломж олгоно. Мөн утга буцаагдаггүй бөгөөд `void` гэсэн түлхүүр үгийг ч хэрэглэдэггүй. Объектийг байгуулах үед тохирох байгуулагчийг нь систем нэг удаа дуудаж хэрэглэдэг. Харин нэгэнт байгуулагдсан объектэд дуудаж хэрэглэхгүй. Класст ямар нэг байгуулагч функцийг илээр тодорхойлж өгөөгүй тохиолдолд `C++ compiler` хамгийн бага хэмжээ бүхий анхдагч функцийг дуудна.
- **Устгагч функц**
 - Ямар нэгэн байгуулагч эсвэл устгагч функцгүйгээр объект үүсч эсвэл устахгүй. Цаашид хэрэглэгдэхгүй объектыг устгах түүний эзэмжшиж байсан ойг чөлөөлөх үүрэгтэй. `Public` шинжтэй байна. Класст ямар нэг устгагч функцийг илээр тодорхойлж өгөөгүй тохиолдолд `C++ compiler` хамгийн бага хэмжээ бүхий устгагч функцийг дуудна. Классын нэртэй ижил нэртэй байна.
- **Зарлах, тодорхойлох**
 - Байгуулагч функц тодорхойлох: Дараах загвараар тодорхойлно.

```
class_name () {  
    strcpy (data0, "");  
    data1 = 0;  
    data2 = 0;
```

...
}

- Устгагч функц тодорхойлох: Зөвхөн өмнөө ~ тэмдэгтэй бичигдэгээрээ ялгагддаг. Employee классын устгагч функцийг ~Employee () гэж зарлаж, дараа нь тодорхойлно. Жишээ нь:

```
~employee () {  
    Cout << “\nObject” << “emp” << k << “is destroyed.”;  
    k--;  
    getch();  
}
```

4. ХЭРЭГЖҮҮЛЭЛТ

1. Класс гэж юу болох, онцлог, үүргийн талаар бич.
 - Класс нь объектийн загвар бөгөөд объектийг уг загварын дагуу бүтээнэ. Класс нь гишүүн өгөгдөл, гишүүн нэгэн цогц нэгдэл юм.
 - Класс дотор зарлагдсан хувьсагчийг классын гишүүн өгөгдөл, класс дотор зарлагдсан функцийг классын гишүүнн функц гэх бөгөөд гишүүн өгөгдөл нь классын шинж чанар, гишүүнн функц нь классын хийх үйлдэл юм. Гишүүн функц нь классын гишүүн өгөгдөлүүдийн харилцах харьцааг хангаж бусад функцуудтэй мэдээлэл солилцдог.
2. Класс болон C++ объектын ялгаа?
 - Класс
 - Класс нь объектийн загвар бөгөөд объектийг уг загварын дагуу бүтээнэ
 - Объект
 - Классын нэг тухайн тохиолдол бөгөөд классаар тодорхойлогддог.
3. Гишүүн функц, гишүүн өгөгдөл хоёр ямар хамааралтай байдаг вэ? Өгөгдлийн битүүмжлэл гэж юу вэ?
 - Гишүүн функц, гишүүн өгөгдөл ямар хамаарал: Объектын өгөгдөлд гишүүн функцээр нь дамжуулж хандах ба объектын өгөгдөлд тухайн классын биш функцээс хандах боломжгүй юм. Иймд гишүүн өгөгдөл санаандгүй үйлдлээс үйлдлээс хамгаалагдана. Өгөгдөл, түүн дээр үйлдэл хийх функцийг хамтад нь холбоотой авч үзэх ОХП – ын нэгэн чухал шинж юм.
 - Битүүмжлэл: Функц, өгөгдлийг тодорхой түвшинд нээлттэй биш объектийн хүрээнд хамтад нь холбоотойгоор авч үздэг ойлголт юм.
4. Классын гишүүн өгөгдөл болон гишүүн функцэд хэрхэн хандах вэ?
 - Классын гишүүн өгөгдөл рүү хандах: Классын private болон protected гишүүн өгөгдөл рүү объектоор нь дамжуулж хандаж болдоггүй. Классын гишүүн өгөгдөлд гишүүн функцээр нь дамжуулж хандана. Гэхдээ бүх

гишүүн өгөгдлийг public гэж тодорхойлж өгвөл ийм өгөгдөлд объектоор нь дамжуулж хандах боломжтой. Энэ тохиолдолд класс гишүүн функтэй байх шаардлагагүй болно.

- ☐ Гишүүн функц рүү хандах: Тухайн классын объектоор нь дамжуулж хандана. Ингэхдээ объектын нэрийг гишүүн функцийн нэрээс шууд сонголтыг цэг (.) оператороор зааглаж бичнэ. Жишээ нь: employee классын getData(), showdata() функцүүдийг emp объектоор дамжуулж дараах байдлаар дууддаг /emp.getData(), emp.getshowdata()/ . Гэхдээ гишүүн функцийн нэрийн өмнө классын нэрийг цэг оператороор зааглаж бичих дүрэм зөрчигдөх тохиолдол байдаг. Гишүүн функц дотроо классынхаа өөр нэг гишүүн функцийг дуудаж хэрэглэж байвал дуудагдаж байгаа гишүүн функцийн нэрийн өмнө ямар ч объектын нэр бичихгүй, ингэж бичих ч боломжгүй юм.

5. Ажилчин гэсэн класс тодорхойлно. Ажилчдын ажилласан цаг бүрийг өөрчилж цалинг тооцоолох жижиг програм бич.

- ☐ class Employee { // өгөгдлүүдийг private байдлаар тодорхойлсон бөгөөд функцүүдийг public байдлаар зарлаж, inline – ээр тодорхойлсон. Ингэснээр зөвхөн тухайн классын функцийг ашиглаж, өгөгдлийг өөрчлөх боломжтой болох юм.
- ☐ private:
- ☐ int empid;
- ☐ char name[20];
- ☐ char position[10];
- ☐ float wtime;
- ☐ public:
- ☐ Employee(){
- ☐ empid = 0;
- ☐ strcpy(name, "");
- ☐ strcpy(position, "Employee");
- ☐ wtime = 0;
- ☐ }
- ☐ void getData(void){
- ☐ cout<<"Enter employee ID: ";
- ☐ cin>>empid;
- ☐ cout<<"Enter name: ";
- ☐ cin>>name;
- ☐ cout<<"Enter position: ";
- ☐ cin>>position;
- ☐ cout<<"Enter work time: ";
- ☐ cin>>wtime;
- ☐ }
- ☐ void showData(){
- ☐ cout<<"Employee ID: "<<empid<<endl;
- ☐ cout<<"Name: "<<name<<endl;

```

❑      cout<<"Position: "<<position<<endl;
❑      cout<<"Work time: "<<wtime<<endl;
❑  }
❑  float calcuSalary(){
❑      if(strcmp(position, "Director") == 0){
❑          return wtime*pricePerHour + dirCalcuSalary();
❑      }
❑      return wtime*pricePerHour;
❑  }
❑  float dirCalcuSalary(){
❑      return wtime*2*pricePerHour;
❑  }
❑  int incWorkTime(float inc){
❑      if(0 < inc && inc <= 24) return 1;
❑      return 0;
❑  }
❑  };

```

5. ДҮГНЭЛТ

Энэхүү лабораторын ажлын асуултуудын хүрээнд 5 – аар бодлого дах классыг байгуулагч болон inline функц ашиглан тодорхойлсон бөгөөд өгөгдлүүдийг private байдлаар функцүүдийг public байдлаар тодорхойл өгсөн. Иймд гишүүн өгөгдөл санаандгүй үйлдлээс үйлдлээс хамгаалагдана. Мөн гишүүн функц дотроо классынхаа өөр нэг гишүүн функцийг дуудаж хэрэглэсэн тул дуудагдаж байгаа гишүүн функцийн нэрийн өмнө ямар ч объектын нэр бичихгүй, ингэж бичих ч боломжгүй байдаг. Үүнээс гишүүн функцийн нэрийн өмнө классын нэрийг цэг оператороор зааглаж бичих дүрэм зөрчигдсөн тохиолдлыг хэрэглэсэн харагдаж байна.

6. АШИГЛАСАН МАТЕРИАЛ

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

7. ХАВСРАЛТ

Кодыг энд оруулна.

Бодлого №5:

1. /* 5. Ажилчин гэсэн класс тодорхойлно. Ажилчдын ажилласан цаг бүрийг өөрчилж цалинг тооцоолох жижиг програм бич.*/
2. #include <iostream>
3. #include <string.h>
4. #define pricePerHour 5000
5. using namespace std;
- 6.

```

7. class Employee {
8. private:
9.     int empid;
10.    char name[20];
11.    char position[10];
12.    float wtime;
13. public:
14.    Employee(){
15.        empid = 0;
16.        strcpy(name, "");
17.        strcpy(position, "Employee");
18.        wtime = 0;
19.    }
20.    void getData(void){
21.        cout<<"Enter employee ID: ";
22.        cin>>empid;
23.        cout<<"Enter name: ";
24.        cin>>name;
25.        cout<<"Enter position: ";
26.        cin>>position;
27.        cout<<"Enter work time: ";
28.        cin>>wtime;
29.    }
30.    void showData(){
31.        cout<<"Employee ID: "<<empid<<endl;
32.        cout<<"Name: "<<name<<endl;
33.        cout<<"Position: "<<position<<endl;
34.        cout<<"Work time: "<<wtime<<endl;
35.    }
36.    float calcuSalary(){
37.        if(strcmp(position, "Director") == 0){
38.            return wtime*pricePerHour + dirCalcuSalary();
39.        }
40.        return wtime*pricePerHour;
41.    }
42.    float dirCalcuSalary(){
43.        return wtime*2*pricePerHour;
44.    }
45.    int incWorkTime(float inc){
46.        if(0 < inc && inc <= 24) return 1;
47.        return 0;
48.    }
49. };

```

```
50.  
51.  
52. int main(){  
53.     Employee e1, e2;  
54.     e1.showData();  
55.     e1.getData();  
56.     e1.showData();  
57.     cout<<"Salary: "<<e1.calcuSalary()<<endl;  
58.     e2.showdata();  
59.     return 0;  
60. }
```