

Байгуулагч, Устгагч Функц (Лаборатори №4)

О. Ихбаяр

ХШУИС, Програм хангамж, 3-р түвшин, 17b1num2575@stud.num.edu.mn

1. ОРШИЛ

Байгуулагч, устгагч функц болон динамик санах ой гэж юу болох тэдгээрийг хэрхэн хэрэглэх болон функц дахин тодорхойлолтын талаар авч үзэх болно.

2. ЗОРИЛГО

Байгуулагч функц, устгагч функц болон анхдагч байгуулагчийн давуу тал, тэдгээрийн хэрэглээний талаарх ойлголтыг авах.

3. ОНОЛЫН СУДАЛГАА

3.1 Байгуулагч функц

Гишүүн өгөгдөлд гарааны утга оноох зорилгоор байгуулагч функцийг хэрэглэдэг. Байгуулагч функц нь классынхаа нэртэй ижил байна.

3.2 Устгагч функц

Устгагч функц нь цаашид хэрэглэгдэхгүй болсон объектыг санах ойгоос чөлөөлөх үүрэгтэй функц юм. Класс дотор устгагч функцийг тодорхойлж өгөөгүй тохиолдолд `c++` компайлэр автоматаар нэмж өгдөг.

3.3 Анхдагч байгуулагч

Анхдагч функц гэдэг нь програм хөгжүүлэгч ямар нэгэн байгуулагч тодорхойлж өгөөгүй байвал `C++` компайлэр өөрөө ийм функцийг програмд нэмж тодорхойлохыг хэлнэ.

3.4 Функц дахин тодорхойлох

Ижил нэртэй бөгөөд параметрых нь төрөл, тоо, дараалал нь өөр функцууд байж болох бөгөөд үүнийг функц дахин тодорхойлох гэнэ.

3.5 NEW оператор

Өгөгдсөн төрлөөр санах ойд шинэ ой нөөцлөөд эхлэл хаягийг нь буцаадаг оператор. Нөөцөлсөн санах ойгоо чөлөөлөхгүй бол төхөөрөмж унтартал устахгүй.

3.6 Устгагч функц

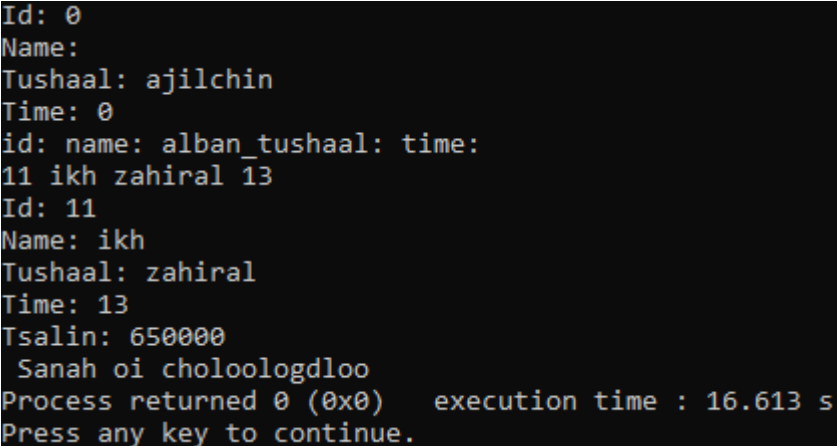
Нөөцөлсөн санах ойг чөлөөлөх үүрэгтэй, классынхаа нэртэй ижил нэртэй ба урдаа `~` тэмдэгтэй байна.

4. ХЭРЭГЖҮҮЛЭЛТ

4.1 Устгагч функц

```
Class employee{
//Устгагч функц тодорхойлох
~employee(){
    cout<<"\n Sanah oi choloologdloo";
}
}
```

Автоматаар дуудагдан ажилсан байдал.



```
Id: 0
Name:
Tushaal: ajilchin
Time: 0
id: name: alban_tushaal: time:
11 ikh zahiral 13
Id: 11
Name: ikh
Tushaal: zahiral
Time: 13
Tsalin: 650000
Sanah oi choloologdloo
Process returned 0 (0x0)   execution time : 16.613 s
Press any key to continue.
```

4.2 Гишүүн функц дотор гишүүн функц дуудах

```
float employee::salary(){
//Tushaal ni zahiraltai tentsuu eschiig shalgana
    int res=strcmp(tushaal,"zahiral");
    if(res==0){
// Zahirliin alban tushaaltai uyd zahirliin tsaling bodoh function duudah
        return time*20000+zah_salary();
    }else{
        return time*20000;
    }
}
```

4.3 Ажилтан классын хүснэгтийг эрэмбэлэх

```
1-r ajiltan
id: name: alban_tushaal: time:
1 ikh zahiral 5
2-r ajiltan
id: name: alban_tushaal: time:
2 bat ajiltan 5
3-r ajiltan
id: name: alban_tushaal: time:
3 dorj ajiltan 6
1-r ajiltan
Id: 2
Name: bat
Tushaal: ajiltan
Time: 5
Tsalin: 100000
2-r ajiltan
Id: 3
Name: dorj
Tushaal: ajiltan
Time: 6
Tsalin: 120000
3-r ajiltan
Id: 1
Name: ikh
Tushaal: zahiral
Time: 5
Tsalin: 250000
```

```
//nootsloh temp object uusgeh
employee temp;
//buble sort ermbelt
for(i=0; i<n;i++){
    for(j=i;j<n-i-1;j++){
        if(emp[j].salary()>emp[j+1].salary()){
            temp=emp[j];
            emp[j]=emp[j+1];
            emp[j+1]=temp;
        }
    }
}
//erembelsen husnegtee hevleh
for(i=0; i<n;i++){
    cout<<"\t"<<i+1<<"-r ajiltan\n";
    emp[i].print();
}
```

BUBLE SORT нь хөрш хоёр элементүүдийн байрыг солих замаар бүх элемент нь зөв байраа олох хүртэлх үйлдлийг хийдэг алгоритм юм.

4.4 Динамик санах ой болон байгуулагч, устагч функцийг хамтад нь ашиглах

Байгуулагч функц дотроо динамик санах ой үүсгэнэ. Устагч функц дотроо delete оператороор нөөцөлсөн санах ойгоо чөлөөлж өгнө.

```
class R{
    char *name;
public:
    R(){
        name = new char[8];
        strcpy(name,"ikhee");
        cout<<name<<"\n";
    }
    ~R(){
        delete name;
        cout<<"ustlaa";
    }
}
```

```
ikhee
ustlaa
Process returned 0 (0x0)    execution time : 0.372 s
Press any key to continue.
```

5. ДҮГНЭЛТ

Гишүүн функц дотор гишүүн функцийг дуудаж болох бөгөөд устгагч функц нь програм ажилаад дуусах үед нөөцөлсөн санах ойг чөлөөлж өгдөг. Объект санах ойгоос чөлөөлөгдөх үед автоматаар дуудагдана.

6. АШИГЛАСАН МАТЕРИАЛ

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

7. ХАВСРАЛТ

```
#include<iostream>
#include<string.h>
using namespace std;
// Ajilchin class todorhoiloh
class employee {
//private handaltiin tuvshintei gishuun ogogdol bolon function
private:
int id;
char name[20];
char tushaal[20];
float time;
float zah_salary();
//public handaltiin tuvshintei gishuun function
public:
void setdata();
void getdata();
void print();
float salary();
bool increase_time(float i);
//Baiguulagch fuction todorhoiloh
employee(){
id=1;
time=0;
strcpy(tushaal,"zahiral");
}
//Ustgagch function todorhoiloh
~employee(){
cout<<"\n Sanah oi choloologdloo";
}
};
main(){
```

```

    int i, n=3,j,bair;
//Array уусgeh
    employee emp[n];
//Utga орuulah heseg
    for(i=0; i<n;i++){
        cout<<"\t"<<i+1<<"-r ajiltan\n";
        emp[i].getdata();
    }
//Noots object уусgeh
    employee temp;
//Tsalingaar ni erembleh
    for(i=0; i<n;i++){
        for(j=i;j<n-i-1;j++){
            if(emp[j].salary()>emp[j+1].salary()){
                temp=emp[j];
                emp[j]=emp[j+1];
                emp[j+1]=temp;
            }
        }
    }
//Tsalingaar ni eremblegdsenii daraa delgetsруу hevleh
    for(i=0; i<n;i++){
        cout<<"\t"<<i+1<<"-r ajiltan\n";
        emp[i].print();
    }
}
//Garaanii utga onоoh
void employee::setdata(void){
    id=0;
    strcpy(name,"");
    strcpy(tushaal,"ajilchin");
    time=0;
}
//Garaas utga орuulah
void employee::getdata(){
    cout<<"id: name: alban_tushaal: time:\n";
    cin>>id>>name>>tushaal>>time;
}
//Delgetsруу hevleh
void employee::print(){

```

```

cout<<"Id: "<<id<<endl;
cout<<"Name: "<<name<<endl;
cout<<"Tushaal: "<<tushaal<<endl;
cout<<"Time: "<<time<<endl;
cout<<"Tsalin: "<<salary()<<"\n";
}
//Tsalin bodoh
float employee::salary(){
    int res=strcmp(tushaal,"zahiral");
    if(res==0){
        return time*20000+zah_salary();
    }else{
        return time*20000;
    }
}
//Zahiraliin tsalin bodoh
float employee::zah_salary(){
    return 30000*time;
}
//Tsag nemegduuleed tuuniigee shalgah
bool employee::increase_time(float i){
    time=time+i;
    if(time>=0 && time<=24){
        true;
    }
    false;
}

```