# Байгуулагч, Устгагч Функц (Лаборатори №5)

О. Ихбаяр

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

# 1. ОРШИЛ

Байгуулагч, устгагч функц болон хуулагч функц гэж юу болох тэдгээрийг хэрхэн хэрэглэх болон онолын тодорхойлолтуудын талаар авч үзэх болно.

# 2. ЗОРИЛГО

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

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 Байгуулагч функц

Гишүүн өгөгдөлд гарааны утга оноох зорилгоор байгуулагч функцыг хэрэглэдэг. Байгуулагч функц нь классынхаа нэртэй ижил байх бөгөөд объект үүсэх үед автоматаар дуудагддаг.

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

## 3.3 Хуулагч функц

С++ компайлер объект үүсэх бүрт хуулагч байгуулагчийг бэлдэж өгдөг учраас хуулагч байгуулагчийг заавал тодорхойлж өгөх шаардлагагүй бөгөөд өмнө нь байгуулагдсан объектын сүүлийн утгыг шинээр үүсэх объектод хуулж авах үүрэгтэй байдаг.

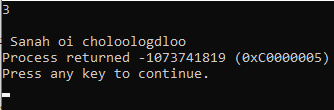
## 3.4 Объектын хаяган хувьсагч зарлах

Class\_name \*object\_pointer\_name хэлбэрээр объектын хаяган хувьсагчийг байгуулах ба түүний гишүүн өгөгдөл болон гишүүн функцэд ‘->’ оператороор хандана. New оператороор санах ой нөөцлөх бөгөөд энэ оператороор санах бэлдсэн үед компьютер унтрах хүртэл санах ойг чөлөөлдөггүй. Үүнээс сэргийлэхийн тулд delete операторыг хэрэглэдэг.

# 4. ХЭРЭГЖҮҮЛЭЛТ

## 4.1 Объектын хаяган хувьсагчийн хүснэгт үүсгэх жишээ

void print(employee emp){

 cout<<emp.getId()<<" "<<emp.getName()<<endl;

Зураг 5 .

}

main(){

employee \*temp;

temp->setId(3);

print(\*temp);

}

## 4.3 Ажилтан классын хүснэгтийг нэрээр эрэмбэлэх

Зураг 5 .

STRCMP(“TEXT”,”TEXT1”) гэх string.h толгой файлын функцийг ашигласнаараа хоёр текст файлыг нэрээр нь жиших боломжтой болж байгаа юм. Үүнийг bubble sort дээр хэрхэн ашигласан байдлыг Зураг 5 2.-т харуулав.

# 5. ДҮГНЭЛТ

Энэхүү лабораторын ажиллаар анхдагч байгуулагч функц, устгагч функц зэрэг зүйлсийг хэрхэн ашиглах талаар мэдэж авсан бөгөөд классын гишүүн өгөгдлийн тусгаарлалтын түвшингээс шалтгаалан хэрхэн хандаж болох вэ гэдгийг мэдэж авсан.

# 6. АШИГЛАСАН МАТЕРИАЛ

1. Объект хандлагат технологийн С++ 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;

char \*tushaal;

float time;

float zah\_salary();

public:

float salary();

bool increase\_time(float i);

//parametrtei baiguulagch function

employee(int id, char name[], char tushaal[], int time){

id=id;

int count=0;

//name-d sanah oi nuutsulj baina.

for(int i=0; i<strlen(name); i++){

count++;

}

name=new char(count);

strcpy(name,name);

count=0;

//tushaal-d sanah oi nuutsulj baina.

for(int i=0; i<strlen(tushaal); i++){

count++;

}

tushaal=new char(count);

strcpy(tushaal,tushaal);

time=time;

}

//Baiguulagch fuction todorhoiloh

employee(){

id=0;

time=0;

name = new char[1];

strcpy(name,"");

string ajiltan ="ajiltan";

tushaal = new char(ajiltan.size());

strcpy(tushaal,"ajiltan");

cout << "Garaanii utga amjilttai orloo"<<endl;

}

//Ustgagch function todorhoiloh

~employee(){

cout<<"\n Sanah oi choloologdloo";

delete[] name;

delete[] tushaal;

}

//utga oruuulah

void setId(int id2){

id=id2;

}

void setName(char name1[]){

//dynamic sanah oi nootsloh

name=new char(strlen(name1));

strcpy(name,name1);

}

void settushaal(char tushaal1[]){

tushaal=new char(strlen(tushaal1));

strcpy(tushaal,tushaal1);

}

void setTime(int time1){

time=time1;

}

//Utga butsaah

int getId(){

return id;

}

char\* getName(){

return name;

}

char\* gettushaal(){

return tushaal;

}

int getTime(){

return time;

}

void print(){ //gishuun ugugdluudiig hewlej baina

cout << "dugaar - " << id << " ner-" << name << " alban tushaal-" << tushaal << " work hour -" << time<<endl;

}

};

main(){

int i, n,j,count=0;

cout<<"Ajiltanii toog oruul"<<endl;

cin>>n;

//Array uusgeh

employee emp[n],\*em[n];

//Utga oruulah heseg

for(i=0; i<n; i++){

cout<<i+1<<"-r hun:\n"<<"id: name: tushaal: time: \n";

int id, time;

char\* name=new char[20];

char\* tushaal=new char[10];

cin>>id>>name>>tushaal>>time;

while(1){

count=0;

for(j=0; j<i; j++){

if(emp[j].getId()==id){

count++;

cout<<"ID dawhtslaa oor id oruulna uu"<<endl;

cin>>id;

}

}

if(count==0){

break;

}

}

emp[i].setId(id);

emp[i].setName(name);

emp[i].settushaal(tushaal);

emp[i].setTime(time);

//emp objectiin haygiig em objectod hadgalah

em[i]=&emp[i];

}

//Bubble sort nereer erembleh

cout << "\n Sorted by names \n";

employee \*temp;

for(int i=0; i<n; i++) { // ajilchdiig nereer ni bubble sort ashiglan erembelj baina

for(int j=1; j<n-i; j++){

if(strcmp(em[j-1]->getName(), em[j]->getName()) > 0){

temp = em[j-1];

em[j-1]= em[j];

em[j]=temp;

}

}

}

//Erembelsnee hevleh

for(i=0; i<n; i++){

em[i]->print();

}

}

float employee::salary(){

int res=strcmp(tushaal,"zahiral");

if(res==0){

return time\*20000+zah\_salary();

}

else{

return time\*20000;

}

}

float employee::zah\_salary(){

return 30000\*time;

}

bool employee::increase\_time(float i){

time=time+i;

if(time>=0 && time<=24){

true;

}

false;

}