**ХУУЛАГЧ БАЙГУУЛАГЧ БА ХУУЛАГЧ ФУНКЦ,**

**ОБЪЕКТЫН ХАЯГАН ХУВЬСАГЧИЙН ТАЛААРХ ОЙЛГОЛТ**

**(ЛАБОРАТОРИ №5)**

Э.Уранчимэг

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

1. **ОРШИЛ**

Энэ тайлангийн онолын судалгаа хэсэгт байгуулагч функц, устгагч функц хэзээ дуудагддаг талаарх асуултад хариулсан. Мөн хуулагч байгуулагч түүний ач холбогдол, хуулагч функц түүний ач холбогдол, энэ нь хэрхэн ойн цоорхойгоос сэргийлдэг тухай дэлгэрэнгүй тайлбарласан. Объектын хаяган хувьсагчийг хэрхэн зарладаг, new оператороор санах ой нөөцлөх, хаяган хувьсагчаар дамжуулж объектын гишүүн өгөгдөл, гишүүн функцэд яаж ханддагийг харуулж, жишээгээр тайлбарласан. Объектын хаяган хувьсагчийн хүснэгт үүсгэх жишээ код бичиж ажиллуулсныг харуулсан. Хэрэгжүүлэлт хэсэгт лаб4 - д тодорхойлсон классын нэр:char[20], албан тушаал:char[10] гэсэн гишүүн өгөгдлийг хаяган хувьсагч болгон өөрчилж гараас өгсөн тэмдэгтийн цуваатай яг ижил урттай санах ой new оператор ашиглан нөөцөлдөг болгосон ба ажилчид нэртэй класс хүснэгт үүсгэж, нэрээр нь эрэмбэлэн, хувийн дугаарыг нь давхардалгүй болгосон кодыг туршиж харуулсан. Мөн хэрхэн хэрэгжүүлснийг тайлбарласан. Хавсралт хэсгээс дэлгэрэнгүй кодыг унших боломжтой.

1. **ЗОРИЛГО**

Классын гишүүн өгөгдөлийг хаяган хувьсагч болгож, санах ойг динамикаар үүсгэх ба объектын хаяган хүснэгттэй ажиллаж, түүнийг эрэмбэлж хэрэгжүүлсэн. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Классын нэр болон тушаал хадгалах гишүүн өгөгдлийг хаяган хувьсагч болгон өөрчлөх,
2. Тухайн хувьсагчийн санах ойг динамикаар байгуулах, устгах,
3. Гишүүн өгөгдлүүдэд гараагны утга оноох,
4. Анхдагч байгуулагчийг тодорхойлох,
5. Параметртэй байгуулагчийг тодорхойлох,
6. Copy функцийг тодоройлох,
7. **ОНОЛЫН СУДАЛГАА**

*3.1 Байгуулагч функц хэзээ дуудагддаг вэ?*

Байгууллагч функц нь объект байгуулах үед автоматаар нэг удаа дуудагддаг.

*3.2 Устгагч функц хэзээ дуудагдах вэ?*

Устгагч функц нь автоматаар объект устах үед нэг удаа комплятораар дуудагддаг. Комплятор нь exe файлыг ажиллуулдаг cpu дээр байдаг зүйл юм.

*3.3 Хуулагч байгуулагч гэж юу вэ? Ач холбогдол нь юу вэ?*

Шинэ объект өөрийнхөө өмнөх аль нэг объектын сүүлийн утгыг хуулбарлаж авах боломжийг хуулагч байгуулагч олгодог. Шинэ объект, хуулбарлах объект хоорондоо нэрээрээ ялгаатай байна. Хуулагч байгуулагч нэг параметр авах ба тэр нь класс нэгтэй хуулагдах объектын заалт байна.

*3.4 Хуулагч функц гэж юу вэ? Ач холбогдол нь юу вэ? Санах ойн цоорхойгоос хэрхэн сэргийлэх вэ?*

Хуулагч байгуулагчаас ялгаатай нь хуулагч функцийг объект байгуулсны дараа нэг объектын утгыг нөгөө объект руу хуулахад ашиглана. Нэг объектын утгыг нөгөө объект руу хуулахдаа хувьсагч тус бүрт нь тэнцүү оператор ашиглаж утга оноосноос copy функцийг ашигласнаар санах ойн цоорхойгоос сэргийлнэ.

*3.5 Объектын хаяган хувьсагчийг хэрхэн зарлах вэ? new оператороор санах ой нөөцлөх, хаяган хувьсагчаар дамжуулж объектын гишүүн өгөгдөл, гишүүн функцэд яаж хандах вэ?*

*Объектын хаяган хувьсагчийг зарлах:* employee \*a;

*New оператороор санах ой нөөцлөх*: а = new employee();

*Хаяган хувьсагчаар дамжуулж гишүүн өгөгдөлд хандах:* a->гишүүн өгөгдөл;

(\*a).гишүүн өгөгдөл;

*Хаяган хувьсагчаар дамжуулж гишүүн функцэд хандах:* a->гишүүн функц;

(\*a).гишүүн функц();

*3.6 Объектын хаяган хувьсагчийн хүснэгт үүсгэх жишээ код бичиж ажиллуул.*

main(){

//i buhel toon turliin huvisagch zarlasan

int i,j;

employee emp[3], \*e[3];

//ehnii objectiig

for(i=0; i<3; i++){

cout<<"Ajiltnii id-g oruul : \n";

start:

cin>>id;

if(i == 0){

emp[i].set\_id(id);

}else{

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

if (id == emp[j].get\_id()){

cout << "Id davhatssan tul dahin Id-g oruul : "<<endl;

goto start;

}

}

emp[i].set\_id(id);

}

cout<<"Ajiltnii neriig oruul : \n";

cin>>name;

cout<<"Ajiltnii alban tushaaliig oruul : \n";

cin>>position;

cout<<"Ajiltnii ajillasan tsagiig oruul : \n";

cin>>time;

emp[i].set\_name(name);

emp[i].set\_position(position);

emp[i].set\_hour(time);

// emp objectiin haygiig objectiin haygan husnegt e-d hadgalah

e[i]=&emp[i];

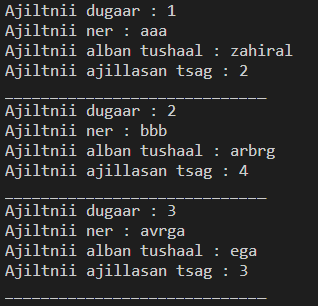
}

for(i=0; i<3; i++){

emp[i].print();

}

Үр дүн:



1. **ХЭРЭГЖҮҮЛЭЛТ**

Лаб04 - д тодорхойлсон классын нэр:char[20], албан тушаал:char[10] гэсэн гишүүн өгөгдлийг хаяган хувьсагч болгон өөрчилж гараас өгсөн тэмдэгтийн цуваатай яг ижил урттай санах ой new оператор ашиглан нөөцөлдөг болго.

class employee{

// private handaltiin tuvshin

private:

//gitsuun ugugdluudiin zarlalt

int dugaar;

char \*ner;

char \*tushaal;

float tsag;

float ceo\_salary();

Үүний тулд анхдагч болон параметертэй байгуулагч функцууд тодорхойлж гишүүн өгөгдөлд гарааны утга онооно. Мөн объект устах үед дээрх хоёр гишүүн өгөгдөлд нөөцөлсөн санайх ойг чөлөөлдөг болгож өөрчил.

//anhdagch baiguulagch funts ene ni umnuh lab deer todorhoilson garaanii utga onooh functstei ijil uildel hiij bga

employee::employee(){

dugaar=0;

//ner huvisagchid dinamikaar sanah oi nuutsluj baina

ner = new char[20];

strcpy(ner, "default");

tushaal = new char[10];

strcpy(tushaal, "ajilchin");

tsag=0;

}

//parametertei baiguulagch functs ene ni umnuh lab deer todorhoilson hereglegchees utga avah functstei ijil uildel hiine

employee::employee(int d, char \*n, char \*t, float ts){

dugaar=d;

ner = new char[strlen(n)+1];

strcpy(ner, n);

tushaal = new char[strlen(t)+1];

strcpy(tushaal, t);

tsag=ts;

}

//ustgagch funktsiin todorhoilolt

employee::~employee(){

delete ner;

delete tushaal;

cout<<"Object deleted;"<<endl;

}

1. Олон ажилчин бүртгэж ажилчдыг нэрээр нь эрэмбэл. Хоёр объектын байрыг солихдоо "copy" функцийг ашиглана.

void sort1(employee emp[] , int a){

int i,j;

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

for(j = i; j > 0; j--){

if(strcmp(emp[j].get\_name() , emp[j-1].get\_name())){

employee tmp;

tmp.copy(emp[j-1]);

emp[j-1].copy(emp[j]);

emp[j].copy(tmp);

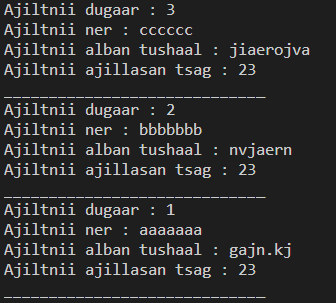
}

}

}

}

Үр дүн:



2. Объектын хаяган хүснэгт үүсгээд хаягийг нь нэрээр нь эрэмбэл.

void sort2(employee \*emp[] , int a){

int i,j;

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

for(j = i; j > 0; j--){

if(strcmp(emp[j]->get\_name(), emp[j-1]->get\_name())>0){

employee \*tmp;

tmp;

tmp=emp[j];

emp[j]=emp[j-1];

emp[j-1]=tmp;

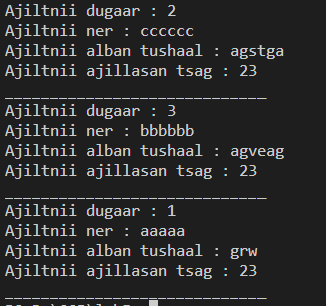
}

}

}

}

Үр дүн:



3. Шинэ ажилчин бүртгэхдээ тухайн ажилчны ID өмнө бүртгэлтэй эсэхийг шалгаж бүртгэлтэй бол бүртгэхгүй. Өөрөөр хэлбэл ID дахин давхцахгүй болго.

cout<<"Ajiltnii id-g oruul : \n";

start:

cin>>id;

if(i == 0){

emp[i].set\_id(id);

}else{

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

if (id == emp[j].get\_id()){

cout << "Id davhatssan tul dahin Id-g oruul : "<<endl;

goto start;

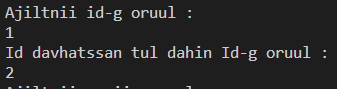
}

}

emp[i].set\_id(id);

}

Давхардсан id оруулсны дараах үр дүн:



1. **ДҮГНЭЛТ**

Энэ лабораторын ажилд хуулагч функцын талаар дэлгэрэнгүй судаллаа. Уг ажлын хүрээнд ажилтны нэрийг 2 янзаар эрэмбэлж үзүүлсэн ба ижилхэн үр дүнг гаргасан.

1. **АШИГЛАСАН МАТЕРИАЛ**

* “Объект хандлагат программчлал лекцийн материал”, М.Золжаргал

1. **ХАВСРАЛТ**

Нэгдсэн кодыг хавсаргав.

#include<iostream>

#include<string.h>

using namespace std;

float undsen\_tsalin=5000;

//employee nertei class zarlasan

class employee{

// private handaltiin tuvshin

private:

//gitsuun ugugdluudiin zarlalt

int dugaar;

char \*ner;

char \*tushaal;

float tsag;

float ceo\_salary();

//public handaltiin tuvshin

public:

//gishuun ugugduld handah set functsuudiin zarlalt

void set\_id(int \_dugaar);

void set\_name(char \*\_ner);

void set\_position(char \*\_tushaal);

void set\_hour(float \_tsag);

void set\_data(int d, char \*n, char \*t, float ts);

//gishuun ugugdliin utgiig avah get funktsuudiin zarlalt

int get\_id();

char \*get\_name();

char \*get\_position();

float get\_hour();

//ajiltnii ugugdliig hevlen haruulah, tsalin bodoh, ajillasan tsag nemegduuleh, ajiltnii medeelliig huulah funktsuudiin zarlalt

void utga\_onooh(employee &a);

void print();

float salary();

bool tsag\_nemegduuleh();

void copy(employee &emp);

//baiguulagch funkts

employee();

//parametertei baiguulagch funktsiin zarlalt

employee(int d, char n[], char t[], float ts);

//ustgagch funktsiin zarlalt

~employee();

};

//anhdagch baiguulagch funts ene ni umnuh lab deer todorhoilson garaanii utga onooh functstei ijil uildel hiij bga

employee::employee(){

dugaar=0;

//ner huvisagchid dinamikaar sanah oi nuutsluj baina

ner = new char[20];

strcpy(ner, "default");

tushaal = new char[10];

strcpy(tushaal, "ajilchin");

tsag=0;

}

//parametertei baiguulagch functs ene ni umnuh lab deer todorhoilson hereglegchees utga avah functstei ijil uildel hiine

employee::employee(int d, char \*n, char \*t, float ts){

dugaar=d;

ner = new char[strlen(n)+1];

strcpy(ner, n);

tushaal = new char[strlen(t)+1];

strcpy(tushaal, t);

tsag=ts;

}

//ustgagch funktsiin todorhoilolt

employee::~employee(){

delete ner;

delete tushaal;

cout<<"Object deleted;"<<endl;

}

//set funktsiig todorhoilson baidal

void employee::set\_id(int \_dugaar){

dugaar=\_dugaar;

}

void employee::set\_name(char\* \_ner){

if(ner != NULL){

delete[] ner;

}

ner = new char[strlen(\_ner)+1];

strcpy(ner, \_ner);

}

void employee::set\_position(char\* \_tushaal){

delete tushaal;

tushaal = new char[strlen(\_tushaal)+1];

strcpy(tushaal, \_tushaal);

}

void employee::set\_hour(float \_tsag){

tsag=\_tsag;

}

//get functionii todorhoilolt

int employee::get\_id(){

return dugaar;

}

char \*employee::get\_name(){

return ner;

}

char \*employee::get\_position(){

return tushaal;

}

float employee::get\_hour(){

return tsag;

}

//delgets ruu printleh print funktsiin todorhoilolt

void employee::print(){

cout<<"Ajiltnii dugaar : "<<dugaar<<endl;

cout<<"Ajiltnii ner : "<<ner<<endl;

cout<<"Ajiltnii alban tushaal : "<<tushaal<<endl;

cout<<"Ajiltnii ajillasan tsag : "<<tsag<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

}

//ajiltnii tsaling bodoh salary functionii todorhoilolt

float employee::salary(){

//tushaal husnegtiin utga zahiral-tai tentsej bval 0-iig butsaah tul 0-tei tentsuu baina uu gj shalgana

if(strcmp(tushaal, "zahiral")==0){

//tiim bol zahirliin tsaling boddog functioniig duudna

return ceo\_salary();

}else{

//ugui bol ajiltnii tsaling bodoj tsalin huvisagchid onoono, tsaling butsaana

return undsen\_tsalin\*tsag;

}

}

//zahirliin tsaling oldog ceo\_salary functionii todorhoilolt

float employee::ceo\_salary(){

float zahiraliin\_tsalin=(undsen\_tsalin+(float)4000)\*tsag;

return zahiraliin\_tsalin;

}

//ajiltnii ajillasan tsagiig nemegduuleh funktsiin todorhoilolt

bool employee::tsag\_nemegduuleh(){

//nemj ajillasan tsagaa hadgalah hours huvisagch zarlaad, huvisagchid hereglegchees utga avna

int hours;

cout<<"heden tsagaar nemegduuleh ve : ";

cin>>hours;

//iluu tsag maani 0-24 tsagiin zavsar baina uu gej shalgana

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

// tiim bol tsagaa nemegduuleed true utga butsaana

tsag=tsag+hours;

return true;

}else

return false;

}

//ajiltnii medeelliig huulah funktsiin todorhoilolt

void employee::copy(employee &emp){

dugaar=emp.get\_id();

if(ner != NULL){

delete[] ner;

}

ner=new char[strlen(emp.ner)+1];

strcpy(ner, emp.get\_name());

if(tushaal != NULL){

delete[] tushaal;

}

tushaal=new char[strlen(emp.tushaal)+1];

strcpy(tushaal, emp.get\_position());

tsag=emp.get\_hour();

}

//ajilchdiig nereer ni erembelj bairiig solihdoo copy function ashiglasan

void sort1(employee emp[] , int a){

int i,j;

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

for(j = i; j > 0; j--){

if(strcmp(emp[j].get\_name() , emp[j-1].get\_name())>0){

employee tmp;

tmp.copy(emp[j-1]);

emp[j-1].copy(emp[j]);

emp[j].copy(tmp);

}

}

}

}

//ajilchdiig neriig ni haygaar ni erembelsen

void sort2(employee \*emp[] , int a){

int i,j;

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

for(j = i; j > 0; j--){

if(strcmp(emp[j]->get\_name(), emp[j-1]->get\_name())>0){

employee \*tmp;

tmp;

tmp=emp[j];

emp[j]=emp[j-1];

emp[j-1]=tmp;

}

}

}

}

main(){

//i buhel toon turliin huvisagch zarlasan

int i,j;

employee emp[3], \*e[3];

//ehnii objectiig

for(i=0; i<3; i++){

cout<<"Ajiltnii id-g oruul : \n";

start:

cin>>id;

if(i == 0){

emp[i].set\_id(id);

}else{

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

if (id == emp[j].get\_id()){

cout << "Id davhatssan tul dahin Id-g oruul : "<<endl;

goto start;

}

}

emp[i].set\_id(id);

}

cout<<"Ajiltnii neriig oruul : \n";

cin>>name;

cout<<"Ajiltnii alban tushaaliig oruul : \n";

cin>>position;

cout<<"Ajiltnii ajillasan tsagiig oruul : \n";

cin>>time;

emp[i].set\_name(name);

emp[i].set\_position(position);

emp[i].set\_hour(time);

// emp objectiin haygiig objectiin haygan husnegt e-d hadgalah

e[i]=&emp[i];

}

for(i=0; i<3; i++){

emp[i].print();

}

while(1){

int p, q;

cout<<"\nAjiltnii tsaling bodoh bol 1-iig songo : \nAjillasan tsagiig nemegduuleh bol 2-iig songo : \nTsalingaar erembeleh bol 3-iig songo : \nNereer ni erembeleh bol 4-iig songo : \nNeriig haygaar ni erembeleh bol 5-iig songo : \nGarah bol 6-iig songo : \n";

cin>>p;

if(p==6){

break;

}

if(p==1){

cout<<"Ajiltnii dugaariig oruul : ";

cin>>q;

cout << "tsalin : "<<emp[q].salary()<< endl;

}else if(p==2){

cout<<"Ajiltnii dugaariig oruul : ";

cin>>q;

emp[q].tsag\_nemegduuleh();

emp[q].print();

}else if(p==3){

sort1(emp,3);

for(i=0; i<3;i++){

emp[i].print();

cout<<emp[i].salary()<< endl;

}

}else if(p==4){

sort1(emp, 3);

for(i=0; i<3; i++){

emp[i].print();

}

}else if(p==5){

sort2(e, 3);

for(i=0; i<3; i++){

e[i]->print();

}

}

}

}