# This хувьсагч ба static гишүүн өгөгдөл (Лаборатори №8)

О. Ихбаяр

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

# 1. ОРШИЛ

This хувьсагч гэж юу болох түүнийг хэрхэн ашигладаг болон ашигласны давуу тал ба яагаад заавал this хувьсагч ашигладаг талаар, түүнчлэн static гишүүн өгөгдөл гэж юу болох түүнийг хэрхэн хэрэглэх талаар авч үзэх болно.

# 2. ЗОРИЛГО

This хувьсгач болон static гишүүн өгөгдөлийг хэрэглэж сурах болон this хувьсагчийн үүргүүдийн талаар тодорхой хэмжээний ойлголттой болж авах зорилготой.

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 this хувьсагч

Классын гишүүн функц дуудагдах үедээ аль объектын гишүүн өгөгдөлд хандах вэ гэдгээ мэдэж байх шаардлагатай. This нь классын гишүүн функцийг дуудаж байгаа объектын хаягийг заах хаяган хувьсагч юм. Комплайтор функцийг дуудахдаа:

1. This нэртэй объектон хаяган хувьсагч зарлана.
2. Тухайн функцийг дуудаж байгаа объектын хаягийг хадгална.
3. Програм counter функцлуу шилжинэ.
4. Тухайн функц дотор ашиглагдсан гишүүн өгөгдлийн өмнө this хувьсагч тавина.

This хувьсагчийг програм эх кодон дээр ил болон далд гэсэн 2 янкаар ашигладаг.

## 3.2 static гишүүн өгөгдөл

Програм ажилж эхлэх үед програм дээр үүсээд прогром ажиллаж дуусахад санах ойг чөлөөлдөг хувьсагчийг static хувьсагч гэнэ. Зөвхөн C++ -ийн хувьд static хувьсагчид гарааны утга оноож өгөх ёстой байдаг dөгөөд static гишүүн өгөгдөл нь классынх ба классаас үүссэн объектууд нь дундаа тэрхүү static хувьсагчийг ашигладаг. Түүнчлэн static гишүүн өгөгдөлд классаар нь хандахаас гадна объктуудаар нь дамжуулан ханддаг. Static гишүүн өгөгдлийг зарлахдаа static хэмээх түлхүүр ашиглан зарладаг. (static int x;)

Static гишүүн өгөгдөлд классаар нь хандахдаа дараах бичиглэлтэй байдаг.(static хувьсагчийн төрөл + класс нэр::static хувьсагч) . Жишээ нь: int A::x=0;

## 3.3 static гишүүн функц

Зөвхөн static гишүүн өгөгдөлрүү хандах боломжтой байдаг бөгөөд түүнийг тодорхойлохдоо мөн static түлхүүр үгийг ашиглан тодорхойлж өгдөг. Жишээ нь:

(static void set\_x(int a){x=a})

# 4. ХЭРЭГЖҮҮЛЭЛТ

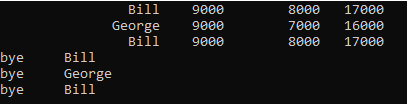
## 4.1 Хуулагч функц тодорхойлох

Тодорхойлолт: e объект байгуулж е1 объектоор гарааны утга оноох үед е болон е1 нь дундаа нэг санах ойг эзэмшихээс сэргийлж хуулагч функцийг зарлаж өгсөн.

Employee e = e1;

// Huulagch function todorhoilj ogoh

employee::employee(const employee &e){

 name =new char[strlen(e.name)+1];

strcpy(name, e.name);

basicpay= e.basicpay;

allowance =e.allowance;

};

## this хувьсагч

void shape2d::set\_a(int a){

Тодорхойлолт: энэ үед a гээд адилхан хоёр хувьсагч яг аль нь объектынх гэдгээ мэдэхээ болино.

a=a;

}

..........................

void shape2d::set\_a(int a){

Тодорхойлолт: энд а нь объект болон параметр гэдгээрээ ялгагдаж өгнө.

this.a=a;

}

## static гишүүн өгөгдөл

shape(){

Тодорхойлолт: static гишүүн өгөгдөл ашиглан туайн классаас хэдэн объект үүссэнийг нь тоолох боломжтой болж байна. Үүний тулд байгуулагч функц дотор static хувьсагчаа нэмэгдүүлээд устгагч функц дотор түүнийгээ нэгээр хорогдуулснаар объект устах үед объект устаж буйг харуулна.

count++;

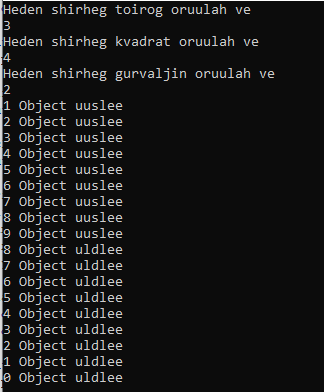
cout<<count<<" Object uuslee"<<endl;

}

~shape(){

count--;

cout<<count<<" Object uldlee"<<endl;

}

# 5. ДҮГНЭЛТ

Энэхүү лабораторын ажиллаар this, static хувьсагчуудыг хэрхэн хэрэглэх талаар болон тэдгээрийн үүргүүдийн талаар тодорхой хэмжээний ойлголттой болсон. Мөн хуулагч функц хэрэглээг мэдэж авсан.

# 6. АШИГЛАСАН МАТЕРИАЛ

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

# 7. ХАВСРАЛТ

## 7.1 shape удамшуулах

#include<iostream>

#include<math.h>

using namespace std;

class shape {

public:

int \*y;

int \*x;

//hiisver function zarlah

static int count;

shape(){

count++;

cout<<count<<" Object uuslee"<<endl;

}

~shape(){

count--;

}

static void set\_count(int a){

count = a;

}

static int get\_coutn(){

return count;

}

virtual int pr()=0;

};

int shape::count=0;

// shapees 2d shape udamshih

class shape2d:public shape {

// a ni taliin urt

public:

string name;

int a;

//hiisver function zarlah

virtual float area()=0;

void set\_a(int a);

};

void shape2d::set\_a(int a){

this.a=a;

}

int shape::pr(){

return 0;

}

class Square:public shape2d{

public:

float area();

int pr();

void tal();

Square(){

x = new int[4];

y = new int[4];

}

~Square(){

delete []x;

delete []y;

}

};

void Square::tal(){

this.x[1]=this.x[0]+this.a;

this.y[1]=this.y[0];

this.y[2]=this.y[0]+this.a;

this.x[2]=this.x[0];

this.y[3]=this.y[0]+this.a;

this.x[3]=this.x[0]+this.a;

}

float Square::area(){

return this.a\*this.a;

}

int Square::pr(){

return this.a\*4;

}

class Circle : public shape2d {

public:

Circle(){

x = new int[1];

y = new int[1];

}

~Circle() {

delete []x;

delete []y;

}

float area();

int pr();

};

float Circle::area(){

return 3.14\*this.a\*this.a;

}

int Circle::pr(){

return 2\*3.24\*this.a;

}

class Triangle : public shape2d {

public:

Triangle(){

x = new int[3];

y = new int[3];

}

~Triangle(){

delete []x;

delete []y;

}

float area();

int pr();

};

float Triangle::area(){

return (this.a\*this.a\*sqrt(3))/4;

}

int Triangle::pr(){

return this.a\*3;

}

main(){

int tn, kn, gn, i=0, k, j;

cout<<"Heden shirheg toirog oruulah ve"<<endl;

cin>>tn;

cout<<"Heden shirheg kvadrat oruulah ve"<<endl;

cin>>kn;

cout<<"Heden shirheg gurvaljin oruulah ve"<<endl;

cin>>gn;

// Object uusgeh

Triangle gurval[gn];

Circle dugui[tn];

Square kvadrt[kn];

shape2d \*\*durs;

//Durs hadgalah sanah oi nootsloh

durs = new shape2d \*[gn+tn+kn];

//duguinuudaa uusgene

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

cout<<i+1<<"-r toirgiin name, towiin koordinat(x,y), radius(R)-g oruul\n";

cout<<"name=> ";

cin>>dugui[i].name;

cout<<"x=> ";

cin>>dugui[i].x[0];

cout<<"y=> ";

cin>>dugui[i].y[0];

cout<<"R=> ";

cin>>dugui[i].a;

cout<<"talbai ni: "<<dugui[i].area()<<endl;

durs[i]=&dugui[i];

}

k=i;

//kvadratuudaa uusgene

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

cout<<i+1<<"-r kvadratiin name, oroin koordinat(x,y), taliin urt(a)-g oruul\n";

cout<<"name=> ";

cin>>kvadrt[i].name;

cout<<"x=> ";

cin>>kvadrt[i].x[0];

cout<<"y=> ";

cin>>kvadrt[i].y[0];

cout<<"a=> ";

cin>>kvadrt[i].a;

cout<<"talbai ni: "<<kvadrt[i].area()<<endl;

durs[k]=&kvadrt[i];

k++;

}

//kvadratuudaa uusgene

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

cout<<i+1<<"-r gurvaljnii name, oroin koordinat(x,y), taliin urt(a)-g oruul\n";

cout<<"name=> ";

cin>>gurval[i].name;

cout<<"x=> ";

cin>>gurval[i].x[0];

cout<<"y=> ";

cin>>gurval[i].y[0];

cout<<"a=> ";

cin>>gurval[i].a;

cout<<"talbai ni: "<<gurval[i].area()<<endl;

durs[k]=&gurval[i];

k++;

}

shape2d \*temp;

//Talbaigaar ni eremblene

for (i = 0; i < k-1; i++){

for (j = 0; j < k - i - 1; j++){

if ( durs[j]-> area() > durs[j + 1] -> area()){

temp = durs[j];

durs[j] = durs[j+1];

durs[j+1] = temp;

}

}

}

//Talbaigaar erembelsnee delgetsnd haruulna

cout<<"\n\tEremblegdsen baidal:\nName\tArea\n";

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

cout<<durs[i]->name<<"\t"<<durs[i]->area()<<endl;

}

delete []durs;

}

7.2 Хуулагч байгуулагч тодорхойлох

#include<iostream>

#include<string.h>

#include<conio.h>

using namespace std;

// employee class heregjuulelt

class employee {

private:

char \*name;

int basicpay;

int allowance;

public:

employee(const employee &e);

employee(void);

~employee(void);

employee(char \*n, int b, int a);

friend ostream & operator <<(ostream &os, employee &e);

};

// Huulagch function todorhoilj ogoh

employee::employee(const employee &e){

name =new char[strlen(e.name)+1];

strcpy(name, e.name);

basicpay= e.basicpay;

allowance =e.allowance;

};

// Baiguulagch function todorhoilg ogoh

employee::employee(){

name = new char[20];

strcpy(name, "Hello, C++");

basicpay= 0;

allowance =0;

}

// Ustgagch function todorhoiloh

employee::~employee(){

cout<<"bye\t"<<name<<endl;

delete name;

}

// Parametrtei baiguulagch functoin todorhoiloh

employee::employee(char \*n, int b, int a){

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

strcpy(name, n);

basicpay= b;

allowance =a;

}

// delgetsend hewleh

ostream & operator <<(ostream &os, employee &e){

int grosspay = e.basicpay + e.allowance;

os.width(20);

os<<e.name;

os.width(8);

os << e.basicpay;

os.width(12);

os<<e.allowance;

os.width(8);

os<<grosspay;

os<<"\n";

return (os);

}

int main(){

// e1 objectod utga onooh

employee e1 ("Bill", 9000, 8000) ;

cout<< e1;

//e2 objectod utga onooh

employee e2 ("George", 9000, 7000);

cout<<e2;

// e1 objectiig e objectod huulj tawih

employee e=e1;

cout<<e;

return 0;

}