# Объект хандалтат технологийн класс ойлголтыг судалж С++ хэлэнд хэрэгжүүлж ажилтны цалин бодох програм бичих (Лаборатори №3)

Я.Тэмүүжин

ХШУИС, МКУТ, ПХ 16B1SEAS3369, y.usott@gmail.com

# 1. ОРШИЛ

Объект хандалтат технологи ба түүний класс ойлголтыг судлаж тус ойлголтоо ашиглаж хавсралт 7.3т байгаа шаардлагийн дагуу ажилчин классыг тодорхойлж ажилтны цалин бодох програм бичнэ.

# 2. ЗОРИЛГО

Объект, класс ойлголтуудыг судлаж С++ хэлэнд хэрхэн хэрэгжүүлнэ. Объект хандалтад технологийн класс, объект ойлголтуудыг ашиглаж ажилтны цалин бодох програм бичнэ. Үүний тулд дараах зорилтуудыг тавьж ажиллана.

1. Класс ба объектын онолын судалгааг хийнэ.
2. Ажилтан классын гишүүн өгөгдөл, гишүүн функцуудыг зарлаж хандалтын түвшинг тодорхойлж өгнө
3. Ажилтан классын гишүүн функцуудыг тодорхойлно.
4. Ажилтан классын объект үүсгэж хэрэглэгчид хялбар цэсийг хэвлэж харуулна.

# 3. ОНОЛЫН СУДАЛГАА

## Класс гэж юу вэ?

Өгөгдлийн бүтэц бол олон төрлийн өгөгдлийн нэгдэл бөгөөд ийм бүтцэн өгөгдлийг олон төрлийн функцээр боловсруулдаг. Харин класс ийм төрлийн функц болон өгөгдлийн холбоог бүтүүмжилсэн логик гэж үздэг ба ийм класс бол өгөгдөл ба функцийн нэгдэл, хэрэглэгчийн тодорхойлсон зохиомол өгөгдлийн төрөл юм. Объект хандалтат програмчлалыг илэрхийлж буй С++ багц командыг класс гэж болно [1]. Өөрөөр класс бол объектын загвар юм. [2]

Жишээ нь: Машин классыг авч үзье.

Өөр өөр нэр, брэнд бүхий олон машин байдаг ч тэд бүгд 4 дугуй, хурны хязгаар, хурд хэмжигч гэх мэт зарим нийтлэг шинж чанаруудтай байдаг. Тэгхээр машин бол класс харин дугуй, хурдны хязгаар, хурд хэмжигч нь шинж чанар юм. [1]

* Класс бол өөрийн гишүүн өгөгдөл, гишүүн функцтэй хэрэглэгчийн зохиомол төрөл
* Гишүүн өгөгдөл нь хувьсагчид харин гишүүн функц нь гишүүн өгөгдлийг удирдах функцүүд бөгөөд гишүүн өгөгдөл, гишүүн функц нь нийлж классын объектын шинж чанар, үйл хөдлөлийг тодохойлно.
* Дээрх машин классын жишээгээр, гишүүн өгөгдөл нь хурдны хязгаар, хурд хэмжигч ... бөгөөд хурд нэмэх, зогсол хийх ... нь гишүүн функц байна.

Классыг тодорхойлход санах ой нөөцлөхгүй. Энэ нь зөвхөн классан хувьсагч буюу объект үүсгэх логик загвар юм. [2]

**Класс тодорхойлох**

С++ хэлэнд class түлхүүр үгийн араас классын нэрийг бичиж класс зарладаг. Классын эх бие нь багц коммандын хаалт({}) дотор тодорхойлогдох бөгөөд ард нь цэг таслал тавьж төгсгөдөг.

1. class КлассНэр {
2. Хандалтын түвшин; // Private, public , protected байж болно
3. Гишүүн өгөгдлүүд; //Ашиглагдах хувьсагчид
4. Гишүүн функцүүд; // Гишүүн өгөгдлүүдтэй ажиллах аргууд (гишүүн өгөгдлүүдэд хандах функцүүд )
5. }; // Класс цэг таслалаар төгсөнө.

## Класс болон С++ объектын ялгаа

Класс нь объектын загвар [2] өөрөөр хэлвэл обьект нь классын нэг удаагийн тохиолдол гэж хэлж болно. Жишээ нь барилгийн зураг төсөл бол класс харин жинхэнэ баригдсан барилга бол тухайн классын нэг удаагийн тохиолдол буюу объект юм. Тухайн барилгийн төсөл зургаар хэдэнч барилга барьж болно харин барилга бүрт өөр өөр хүмүүс ажиллаж амьдагдаг байж болно. Класс тодорхойлход санах ой нөөцлөгдөхгүй ба зөвхөн түүний объектийг үүсгэх үед объектод санах ой нөөцлөгддөг.

**Khun класс тодорхойлж байна.**

1. Class Khun {
2. public: String name;private char registerNo[10];String lastName;int phoneNumber;public: void setName(ner) {
3. name = ner;
4. }
5. void setLastName(ovog) {
6. lastName = ovog;
7. }
8. String getLastName() {
9. return lastName;
10. }
11. String getName() {
12. return Name;
13. }
14. }

Khun классын Temuujin объектыг үүсгэж түүний гишүүн функцуудыг ашиглаж гишүүн өгөгдөлд утга оноож байна.

1. Khun Temuujin;
2. Temuujin.setName = ”Temuujin”;
3. Temuujin.setLastName = ”UNKHOWN”
4. Cout<<Temuujin.getLastName()<<” ”<<Temuujin.getName()<<endl;

Үр дүн

1. UNKHOWN Temuujin

## Гишүүн функ, гишүүн өгөгдөл хоёр ямар хамааралтай байдаг вэ?

Классын гишүүн өгөгдөл тус классын гишүүн функцээр дамжуулж хандана. Жишээ нь private хандалтын түвшинтэй гишүүн өгөгдөл нь тус утгийг хэрэглэгчрүү буцаадаг ба ирсэн утгийг тус өгөгдөл оноодог хоёр хувьсагчтай байж болно гэсэн үг. Гишүүн өгөгдөл нь мэдээллийг хадгалдаг бол гишүүн функц нь гишүүн өгөгдлийн мэдээлэл дээр боловсруулалт хийж бусад гишүүн функцтэй мэдээлэл солилцдог [2]. Онолын судалгаа Khun классын тодорхойлолтын жишээг харвал гишүүн өгөгдөл name, lastName –н өгөгдлийг авах болон оноох getName, setName, getLastName, setLastName гэсэн функууд байгаан тухайн гишүүн функцууд нь гишүүн өгөгдлийн утга дээр ажиллаж байгааг харуулна.

## Өгөгдлийн битүүмжлэл гэж юу вэ?

Өгөгдлийн битүүмжлэл гэдэг нь объектын утга далд байна. Энэ нь түүнийг бусдаас хамгаалах нууц гэсэн санаа биш бөгөөд обьектын гишүүн өгөгдөлд зөвхөн гишүүн функцээр нь дамжуулж хандахыг хэлнэ. Объектын өгөгдөлд тухайн классын биш функцээс хандаж болохгүй. Ингэснээр гишүүн өгөгдөл санаандгүй үйлдлээс хамгаалагдана. Өгөгдөл түүн дээр үйлдэл хийх функцыг хамтад нь холбоотой авч үздэг нь объект хандалтат технологийн нэгэн чухал шинж юм. Класс гэх цогч ойлголтонд өгөгдөл функц нь нэгдэн хамаарагддаг. [2]

#include<iostream>

using namespace std;

class ExampleEncap{

private:

/\*Гишүүн өгөгдлүүдийг private байхаар өгсөн тул классын гадна талаас гишүүн

өгөгдөлрүү шууд хандах боловжгүй болж байна.Хандах бол get, болон

set Функцийг дуудах хэрэгтэй

\*/

int num;

char ch;

public:

/\* Get функц нь гишүүн өгөгдлийн утгийг авах зориулалттай.

Функцууд нь public хандалтын түвшинтэй байгаа тул классын гаднаас хандах боломжтой

Энэ нь тэднээр дамжуулан гишүүн өгөгдөлд хандах боломжыг олгоно.

\*/

int getNum() const {

return num;

}

char getCh() const {

return ch;

}

/\* Set функцууд нь private хандалтын түвшинтэй гишүүн өгөгдлүүдэд утга оноох үүрэгтэй

\*/

void setNum(int num) {

this->num = num;

}

void setCh(char ch) {

this->ch = ch;

}

};

int main(){

ExampleEncap obj;

obj.setNum(100);

obj.setCh('A');

cout<<obj.getNum()<<endl;

cout<<obj.getCh()<<endl;

return 0;

}

[3]

Үр дүн

100

A

Гэж гарч байгаа нь set,get функцууд нь ажиллаж байгааг харуулж байна.

## Классын гишүүн өгөгдөл болон гишүүн функцэд хэрхэн хандах вэ?

Классын гишүүн өгөгдөл болон гишүүн функцэд объектын цэг(‘.’) буюу шууд хандалтын оператор ашиглах хандах боломжтой. Жишээ нь: объектын нэр *obj* бөгөөд түүний *printName();*  гишүүн функцэд хандахыг хүсвэл *obj.printName();* гэж бичиж өгөх ёстой.

**Гишүүн өгөгдөлд хандах**

Public хандалтын түвшинтэй гишүүн өгөгдөлрүү өмнө дурьдсанчлан ханддаг бол private гишүүн өгөгдөлд объектоор шууд хандахыг зөврөөшдөггүй. Гишүүн өгөгдөлдөлд хэрхэн хандах нь түүний хандалтын түвшингөөс шууд хамаарна. Хандалтын удирдлага нь С++ хэлэнд хандалтын түвшингээр илэрхийлэгдэнэ/Access modifiers in C++/. Хандалтыг гурван түвшин байдаг: **public, private болон protected.**

1. //гишүүн өгөгдөлд хандах хандалтын илэрхийлэх // с++ програм
2. #include < iostream > using namespace std;
3. **class** Person { // Хандалтын түвшин тодорхойлолт
4. **public**: //гишүүн өгөгдөл
5. string firstName; //гишүүн функц()
6. **void** printfName() {
7. cout << “Minii ner: ” << firstName;
8. }
9. };
10. int main() { //person классын объектыг зарлаж байна
11. Person person1; //гишүүн өгөгдөлд хандаж байна
12. Person1.firstName = “Temuujin”; //гишүүн функцэд хандаж байна
13. person1.printName();
14. **return** 0;
15. }

Үр дүн

Minii ner: Temuujin

**Класс дахь гишүүн функцүүд**

Гишүүн функц тодорхойлох 2 арга байдаг.

* Классын дотор тодорхойлох
* Классын гадна тодорхойлох

Гишүүн функцыг классын гадна зарлахдаа үйлчлэх хүрээний операторыг(::) классын болон функцын нэртэй хамт ашиглах ёстой.

1. #include<iostream>
2. using namespace std;
3. **class** Person {
4. **public**:
5. string fname;
6. **int** id;
8. //printname класс дотор тодорхойлогдохгүй
9. **void** printname();
11. //printid класс тодор тодорхойлогдож байна
12. **void** printid() {
13. cout << ”Person Id = ” << id;
14. }
15. };
16. // үйлчлэх хүрээний оператор(::) ашиглаж printname -г тодорхойлж байна.
17. **void** Person::printname() {
18. cout << ”Person name is: ” << fname;
19. }
21. int main() {
22. Person person1;
23. Person1.fname = ”Temuujin”;
24. Person1.id = 12;
26. //printname()-г дуудаж байна.
27. person1.printname();
28. cout << endl;
29. //printid() -г дуудаж байна.
30. **return** 0;
31. }

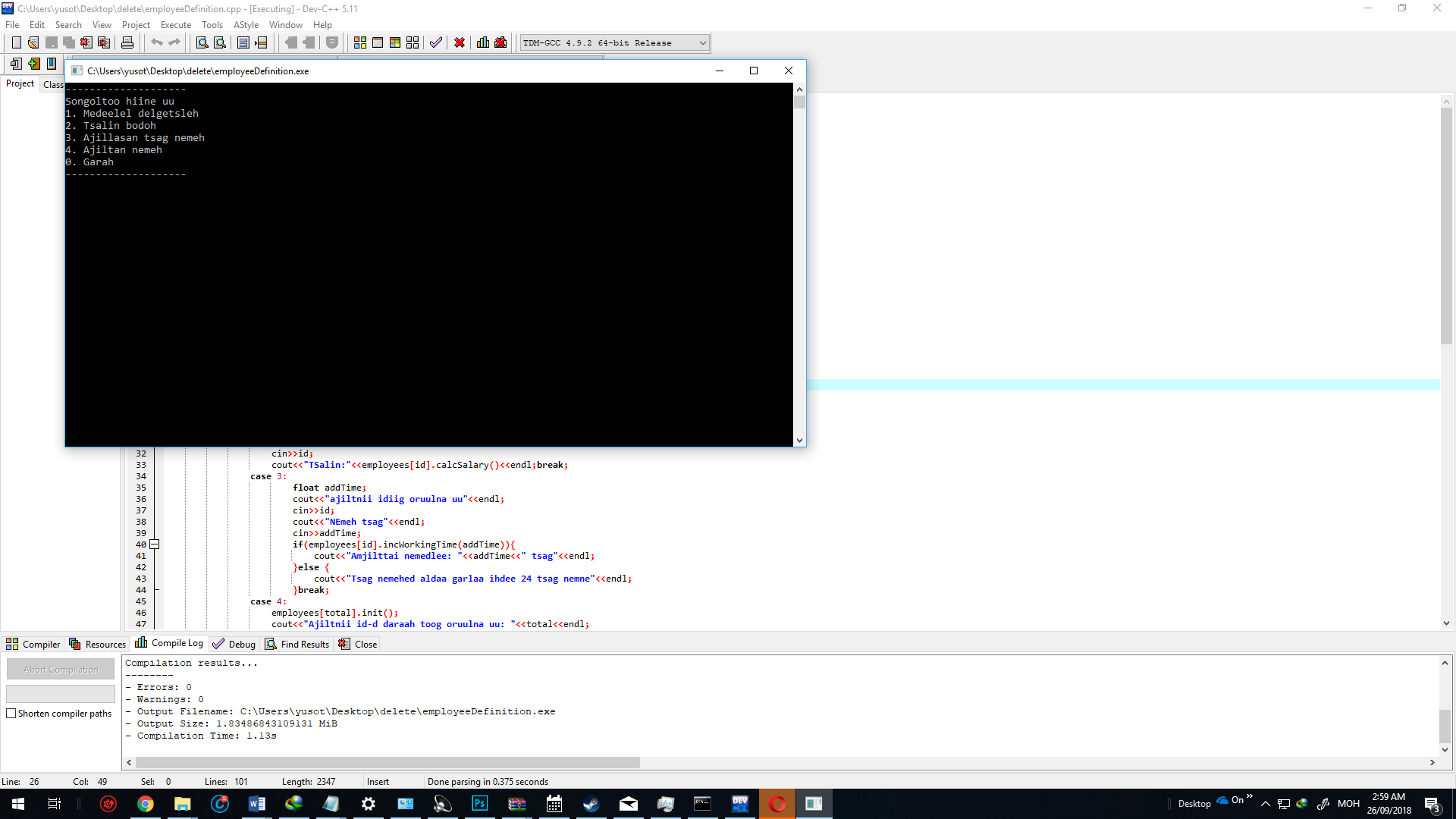
Үр дүн:

1. Person name is: Temuujin
2. Person id: 12

Классын тодорхойлолт дотор тодорхойлогдож буй гишүүн функц нь өгөгдмөл утгаара дотоод мөр функц байдаг гэхдээ мэдээж класс биш ямарч функцийг

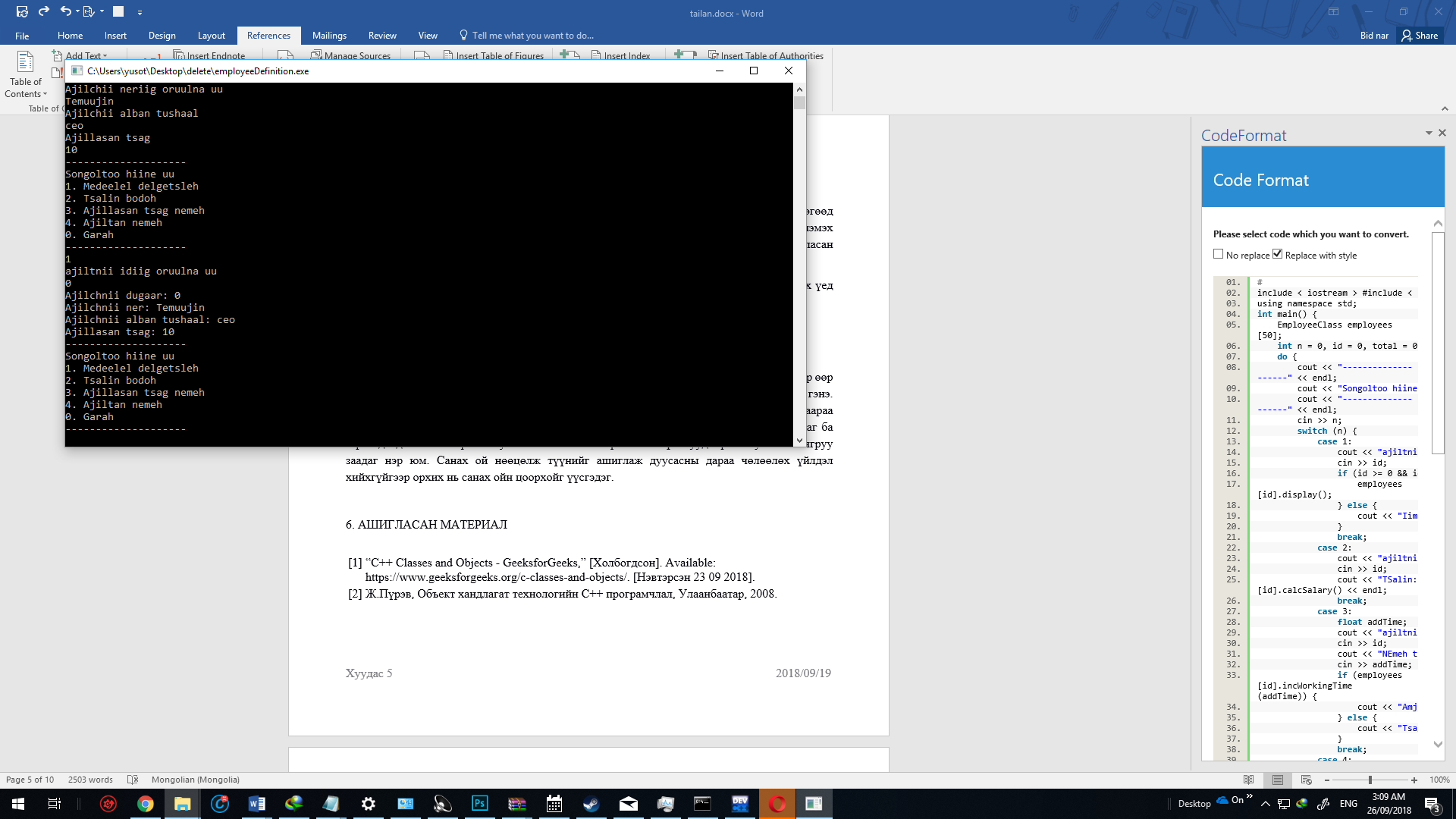
# 4. ХЭРЭГЖҮҮЛЭЛТ

Хавсралт 7.3-н дагуу ажилчин класс болон түүний гишүүн өгөгдөл, гишүүн функцуудыг тодорхойлсон(код хавсралт 7.1 ба 7.2). Тухайн цалингийн програм нь нийт 51 ажилтны мэдээллийг хадгалах бөгөөд мэдээлэл авах, цалин бодох, ажилсан цаг нэмэх, ажилтан нэмэх тухайн функцуудыг дуудах болгонд ажилтны дугаар хэрэг болно. Ажилтны дугаарыг массивын элементийн тоогоор авч явсан. Цалин бодох гишүүн функц нь ажилтаны ажилласан нийт цагийг нэг цагийн хөлсөөр үржүүлж олох бөгөөд үүний дараа ажилтны албан тушаалыг шалгана хэрвээ тухайн ажилтны албан тушаал захирал байвал захиралын цалин бодох функцыг дуудаж түүнээс ирэх хариуг ажилтны цалин бодсон дээр нэмэж захирлын цалинг олно. Хэрэглэгчид амар байх үүднээс программыг сонголттой цэстэй болгосон.

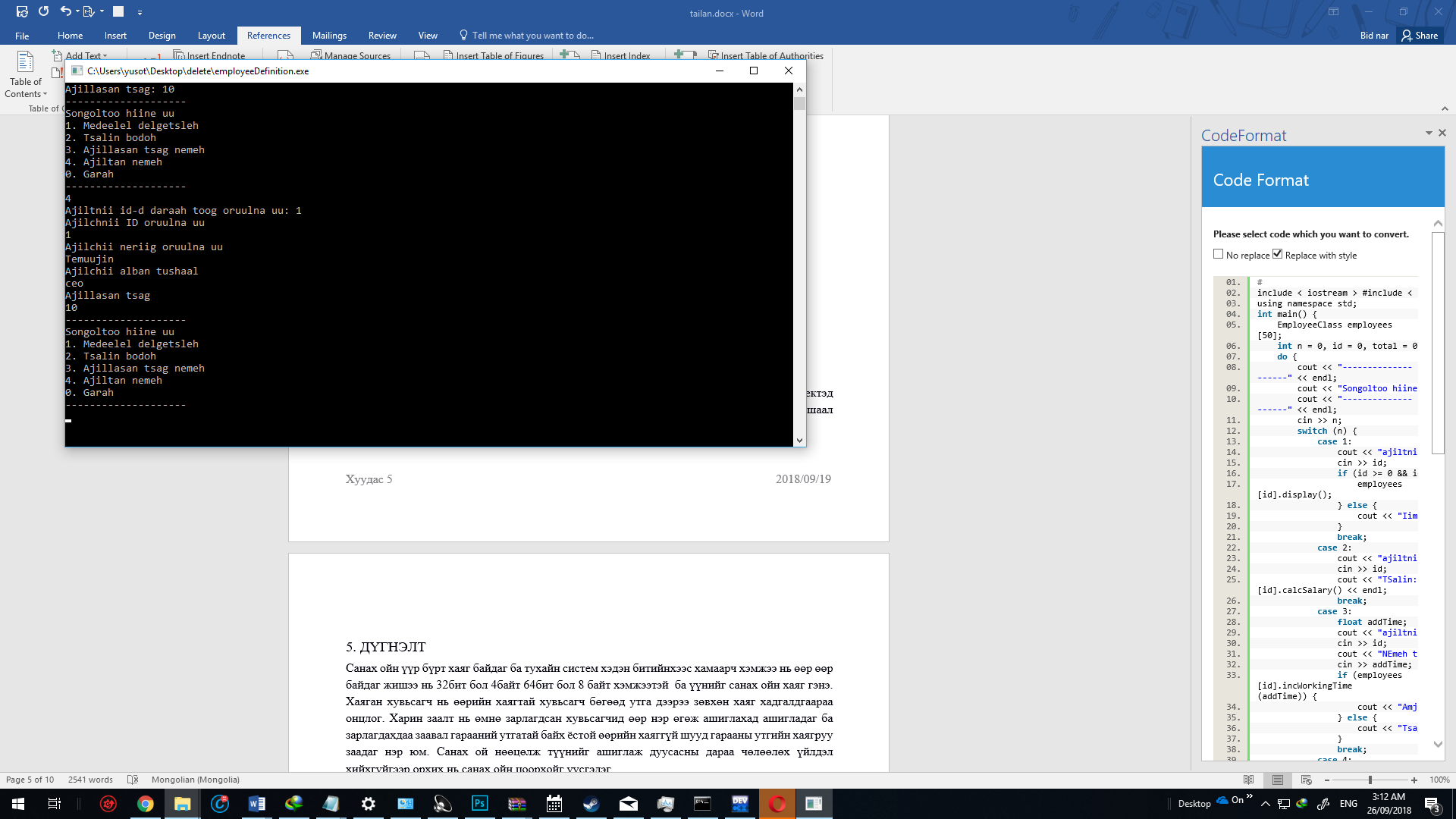


Ажилласан цаг нэмэх функц нь параметрээр нэмэх цагийн хэмжээг агуулах бөгөөд параметрээр авахдаа заалтыг ашигласан нь санах ойг хэмнэж байгаа юм. Ажилласан цаг нэмэх функ нь параметрээр орж ирсэн утга 24-с бага байгаа эсэхийг шалгаах үнэн байвал ажилласан цаг дээр нэмэх гэж буй цагийг нэмээд 1-г буцаана харин 24-с байвал 0-г буцаана.

Мэдээлэл дэлгэцлэх хэсэгт орход мөн ажилтны дугаарыг асуух бөгөөд гараас утга орох үед тухайн индекс дээр буй объектын мэдээллийг харуулна.



Ажилтан нэмэх функц нь өмнө нь нэмэгдсэн n обьектын дараа буюу n+1 индекс дээрх объектэд утга нэмэх ба хэрэглэгчид ямар ажилтны дугаар ашиглахыг зааварлаж өгнө. Мөн албан тушаал дээр ceo гэж бичвэл тухайн объеътын цалин захирлын цалингаар бодогдоно.



# 5. ДҮГНЭЛТ

Класс нь өөрийн гишүүн өгөгдөл, гишүүн функцтэй хэрэгчийн зохионмол өгөдлийн төрөл мөн өөрөөр объектын загвар бөгөөд классыг тодорхойлход санах ой нөөцлөгддөгүй, харин түүний объектыг үүсгэхэд санах ой нөөцлөгдөнө. Объект нь классын нэг удаагийн тохиолдол юм. Объектын гишүүн өгөгдөл, гишүүн функцэд хэрхэн хандах нь түүний хандалтын түвшингээс хамаардаг. Гишүүн функцыг классын тодорхойлолтын дотор болон гадна зарлаж болох бөгөөд гадна зарлахдаа үйлчлэх хүрээний операторыг(::) классын нэр болно функцын нэртэй хамт ашиглана.

6. АШИГЛАСАН МАТЕРИАЛ

|  |  |
| --- | --- |
| [1] | “C++ Classes and Objects - GeeksforGeeks,” [Холбогдсон]. Available: https://www.geeksforgeeks.org/c-classes-and-objects/. [Нэвтэрсэн 23 09 2018]. |
| [2] | Ж.Пүрэв, Объект хандлагат технологийн С++ програмчлал, Улаанбаатар, 2008. |

# 7. ХАВСРАЛТ

## 7.1 Классын зарлалт, толгой файл. employeeClass.h

1. //EmployeeClass классыг зарлаж байна
2. class EmployeeClass {
3. //Хандалтын түвшинг public байх гишүүн өгөгдлүүдийг тодорхойлж байна
4. public:
5. //ажилтны дугаарыг хадгалах хувьсагч
6. int id;
7. //Ажилтны нэрийг хадгалах хувьсагч
8. char name[20];
9. //Ажилтны албан тушаалыг хадгалах хувьсагч
10. char position[10];
11. //Ажилтны ажилласан хугацааг хадгалах хувьсагч
12. float workTime;
13. //Хандалтын түвшин public байх гишүүн функцуудийг зарлаж байна.
14. public:
15. //Дуудахдах үедээ тухайн объектдоо анхны утга өгөх функц зарлаж байна
16. void init();
17. //Ажилтны мэдээллийн утгыг гараас авах функц зарлаж байна
18. void input();
19. //Ажилтны цалин бодох функц зарлаж байна
20. float calcSalary();
21. //Захирлын цалин бодох функц зарлаж байна
22. float calcCeoaSalary();
23. //Ажилтны мэдээллийг харуулах функц зарлаж байна
24. void display();
25. //Ажилласан хугацааг нэмэх функц зарлаж байна
26. bool incWorkingTime(float & addTime);
27. };

## 7.2 Классын тодорхойлолт ба main функц. employeeDefinition.cpp

1. #include<iostream>
2. #include<string.h>
3. //Классын зарлалт буй толгой файлыг дуудаж байна. Хавсралт 7.1д байгаа код.
4. #include"employeeClass.h"
6. using namespace std;
7. int main() {
8. //EmployeeClass классын 50 объектон массив үүсгэж байна
9. //Тус массивт нэмэгдсэн ажилчдын мэдээллийг хадгална.
10. //Массивын элемент бүрд EmployeeClass-н объект байж болно.
11. EmployeeClass employees[50];
12. //Програмд хэрэглэгдэх хувьсагчдыг захлаж анхны утгийг оноож байна
13. // n нь менюний сонголт хийхэд
14. // id-д хэрэглэгчийн гараах орох ажилчны дугаар хадгална
15. //total нь одоогоор хэдэн ажилчин бүртгэсэн байгааг харуулна.
16. //0-с эхэлж байгаа тул байгаа утга нь 0с их бол дээр нь 1-г нэмэхэд нийт ажилчны тоо гарна
17. int n = 0, id = 0, total = 0;
18. //do-while давталтаар цэс хэвлэж хэрэглэгч 0 оруултал цэсийг давтаж гаргана
19. do {
20. cout << "--------------------" << endl;
21. //хэрэглэгчид цэсний зааврийг харуулж байна.
22. cout << "Songoltoo hiine uu" << endl
23. << "1. Medeelel delgetsleh" << endl
24. << "2. Tsalin bodoh" << endl
25. << "3. Ajillasan tsag nemeh" << endl
26. << "4. Ajiltan nemeh" << endl
27. << "0. Garah" << endl;
28. cout << "--------------------" << endl;
29. //цэсний сонголтыг хадгалах хувьсагчид утга оноож байна
30. cin >> n;
31. //n-нь утгад хамаарах case ажиллана.
32. switch (n) {
33. case 1:
34. //хэрэглэгчид заавар хэвлэж байна
35. cout << "ajiltnii idiig oruulna uu" << endl;
36. //мэдээллийг нь харах ажилтны дугаарыг хэрэглэгчээс авч байна
37. cin >> id;
38. //оруулсан ажилтны дугаар 0-50 ийн хооронд буюу нийт массивийн уртаас хэтрэхгүй байх нөхцөлийг шалгаж байна
39. if (id >= 0 && id <= 50) {
40. // хэрэглэгчийн оруулсан id дахь объектын мэдээллийг хэвлэж байна
41. employees[id].display();
42. }
43. //39-р мөрний нөхцөл худал тохиолдолд алдааны заавар хэвлэнэ.
44. else {
45. cout << "Iim idtai ajiltan bhgui" << endl;
46. }
47. //switch-case ээс гарч байна
48. break;
49. case 2:
50. //хэрэглэгчид заавар хэвлэж байна
51. cout << "ajiltnii idiig oruulna uu" << endl;
52. //мэдээллийг нь харах ажилтны дугаарыг хэрэглэгчээс авч байна
53. cin >> id;
54. //хэрэглэгчийн оруулсан id индекс дахь объектын цалин бодох гишүүн функцийг дуудаж буцаж ирэх утгыг хэвлэж байна
55. cout << "TSalin:" << employees[id].calcSalary() << endl;
56. //switch-case ээс гарч байна
57. break;
58. case 3:
59. //Нэмэх цагийг хадгалах хувьсагчийг хадгалж байна
60. float addTime;
61. //Хэрэглэгчид заавар хэвлэж байна
62. cout << "ajiltnii idiig oruulna uu" << endl;
63. //мэдээллийг нь харах ажилтны дугаарыг хэрэглэгчээс авч байна
64. cin >> id;
65. //Хэрэглэгчид заавар хэвлэж байна
66. cout << "NEmeh tsag" << endl;
67. //Ажилласан цагийг хэдэн цагаар нэмэгдүүлэхийг хэрэглэгчээс авч байна
68. cin >> addTime;
69. // хэрэглэгчийн оруулсан id индекс дахь объектын ажилласан цаг нэмэгдүүлэх гишүүн фунзцруу нэмэх цагаа addTime аа параметрээр дамжуулж байна. Хэрвээ гишүүн функц 1 гэж буцаавал нэмэгдсэн гэж үзнэ хэрвээ 0 байвал нэмэгдээгүй алдаа гарсан гэж үзнэ.
70. if (employees[id].incWorkingTime(addTime)) {
71. //Амжилттай нэмсэн бол мэдэгдэх зурвас нэмсэн цагийг хэвлэж байна.
72. cout << "Amjilttai nemedlee: " << addTime << " tsag" << endl;
73. } else {
74. //алдаатай байсан бол алдааны зурвас хэвлэж байна
75. cout << "Tsag nemehed aldaa garlaa ihdee 24 tsag nemne" << endl;
76. }
77. //switch-case ээс гарч байна
78. break;
79. case 4:
80. //сүүлийн объектийн дараах объектын гарааний утга өгөх гишүүн функцийг дуудаж байна
81. employees[total].init();
82. //Хэрэглэгчид сул байгаа индексийн тоог ажилтны дугаар болгохыг санал болгож байна
83. cout << "Ajiltnii id-d daraah toog oruulna uu: " << total << endl;
84. //Хэрэглэгчийн оруулсан индекс дахь объектэд утга авч байна
85. employees[total].input();
86. //Ажилтан нэмэгдсэн гэж үзэж дараагийн сул объектыг индексийн утга болгож байна
87. total += 1;
88. //switch-case ээс гарч байна
89. break;
90. }
91. //Хэрэглэгчээс n-д 0 утга очих хүртэл цэсийг давтаж харуулна.
92. } while (n != 0);
93. return 0;
94. }
95. //Гарааны утга олгох функцыг тодорхойлж байна
96. void EmployeeClass::init() {
97. //Ажилтны дугаар гарааны утга 0
98. id = 0;
99. //Ажилтны нэрний гарааны утга оноож байна
100. strcpy(name, "Todorhoiloogui");
101. //Албан тушаалын гарааны утга оноож байна
102. strcpy(position, "ajiltan");
103. //Ажилласан цагийн гарааний утга оноож байна
104. workTime = 0;
105. }
106. //Хэрэглэгчээс ажилтны мэдээллийг авах гишүүн функц тодорхойлж байна
107. void EmployeeClass::input() {
108. //хэрэглэгчид заавар хэвлэж байна
109. cout << "Ajilchnii ID oruulna uu" << endl;
110. //ажилтны дугаарыг гараас авч байна
111. cin >> id;
112. //хэрэглэгчид заавар хэвлэж байна
113. cout << "Ajilchii neriig oruulna uu" << endl;
114. //ажилтны нэрийг гараас авч байна
115. cin >> name;
116. //хэрэглэгчид заавар хэвлэж байна
117. cout << "Ajilchii alban tushaal" << endl;
118. //ажилтны албан тушаалийг гараас авч байна
119. cin >> position;
120. //хэрэглэгчид заавар хэвлэж байна
121. cout << "Ajillasan tsag" << endl;
122. //ажилтны ажилласан цагийг гараас авч байна
123. cin >> workTime;
124. }
125. //Ажилтны цалин бодох функц тодорхойлж байна. Бодит тоо буцаана. Нэг цагийн үнэлгээ ажилтных 20000 байна.
126. float EmployeeClass::calcSalary() {
127. //back бүхэл тоон хувьсагч зарлаад ажилтны нэг цагийн хөлсийг 20000төгрөгөөр тооцож байна
128. float back = workTime \* 20000;
129. //тухайн ажилтан захиарал эсэхийг харьцуулалт хийж шалгаж байна
130. if (strcmp(position, "ceo") == 0) {
131. //Захирал бол захиралын цалин бодох фунзкцийг дуудаж буцаж ирэх утгыг back-д нэмэж өгч байна
132. back += calcCeoaSalary();
133. }
134. //функцээс back-г буцааж байна.
135. return back;
136. }
137. //захирлын цалин бодох функцийг зарлаж байна
138. float EmployeeClass::calcCeoaSalary() {
139. //захиралын цагийн цалин 30000төг ба ажилласан цалинг 10000-р үржээд буцааж байна. 132-г мөр дээр захирлын цалин энгийн ажилтнаар бодоод энэ функцээс буцах утгийг нэмэж өгч байгаа
140. return workTime \* 10000;
141. }
142. //Мэдээлэл хэвлэх гишүүн функцийг тодорхойлж байна. Утга буцаахгүй
143. void EmployeeClass::display() {
144. //Ажилтны дугаар хэвлэж байна
145. cout << "Ajilchnii dugaar: " << id << endl;
146. //Ажилтны нэр хэвлэж байна
147. cout << "Ajilchnii ner: " << name << endl;
148. //Ажилтны албан тушаал хэвлэж байна
149. cout << "Ajilchnii alban tushaal: " << position << endl;
150. //ажилтны ажилласан цагийг хэвлэж байна
151. cout << "Ajillasan tsag: " << workTime << endl;
152. }
153. //Ажилтны ажилласан цагийг буцаах гишүүн функцыг тодорхойлж байна. Нэмэх цаг нь ихдээ 24 байх бөгөөд илүү болон сөрөг тоо оруулсан бол алдаагэж үзэж 0-г буцаана. ХАрин зав цаг нэмсэн бол ажилласан цаг дээр гараас оруулсан цагийг нэмээд 1-г буцаана.
154. bool EmployeeClass::incWorkingTime(float & addTime) {
155. //24-с бага буюу тэнцүү эсэхийг шалгаж байна
156. if (addTime <= 24) {
157. //Нийт ажилласан цаг дээр нэмэх цагийг нэмэж байна
158. workTime += addTime;
159. //амжилттай нэмсэн тул 1-г буцааж байна
160. return 1;
161. } else {
162. //156 мөрний нөхцөл худал болсон тул 0-г буцааж байна
163. return 0;
164. }
165. }

## 7.3 Ажилчин классын шаардлага

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

-------------------------------------------------------

Ажилчин

-------------------------------------------------------

+Дугаар : int

+Нэр : char[20]

+Албан тушаал : char[10]

+Ажилласан цаг : float

-------------------------------------------------------

+Гарааны утга оноох() : void // дугаар=0, нэр="", албан тушаал="ажилчин", ажилласан цаг=0

+Гараас утга авах() : void

+Мэдээлэл дэлгэцлэх() : void

+Цалин бодох() : float // хэрэв албан тушаал нь захирал байвал Захирлын цалин бодох() функцийг дуудаж үндсэн цалин дээр нэмнэ

+Захирлын цалин бодох() : float

+Ажилласан цаг нэмэгдүүлэх(float) : boolean /0, 1/ //"Ажилласан цаг"

//гэсэн шинжийг нэмэгдүүлэх бөгөөд 0-оос 24-ийн хооронд утга авч

//байгааг шалгаад хэрэв амжилттай болвол 1, үгүй бол 0 гэсэн утга буцаана

-------------------------------------------------------