# Класс Дотор Санах Ойг Динамикаар Үүсгэж Өгөгдлийг Хадгалах Лабораторийн Ажил

# (Лаборатори №5)

# 1. ОРШИЛ/УДИРТГАЛ

Амьдралд байгаа объектыг компьютерийн програмд загварчлахад классыг тодорхойлж, объект үүсгэн ажилчдыг бүртгэх, цаг бүртгэх, цалин тооцох, нэрээр нь эрэмбэлэх зэрэг үйлдлийг хийнэ. Анхны утгыг байгуулагч функцээр дамжуулж олгоно. Тэмдэгт мөрийг хадгалахдаа динамикаар санах ойг нөөцөлж хадгална.

# 2. ЗОРИЛГО

Ажилчин нэртэй объектын классыг өгөгдсөн шаардлагын дагуу зарлаж, гишүүн функцуудийг тодорхойлно. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Гишүүн өгөгдлүүдийн төрлийг тодорхойлох,

2. Гишүүн функц тодорхойлохдоо тэмдэгт мөр хадгалах гэж байгаа бол санах ойг нөөцлөх,

3. Байгуулагч функц тодорхойлох,

4. Устгагч функц тодорхойлох,

5. Ажилчин классын хүснэгт үүсгэж ажилчдыг бүртгэж боловсруулалт хийх.

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 Байгуулагч функц

Байгуулагч функц нь объектын гишүүн өгөгдөлд дамжуулсан утга эсвэл анхны утгыг өгөх зориулалттай. Энэхүү функцын нэр нь классын нэртэй ижил байна. Обьект үүсэх үед compiler байгуулагч функцыг автоматаар дуудна. Байгуулагч функцын буцаах утгын төрлийг тодорхойлдоггүй. Хэрэв классд байгуулагч функцыг тодорхойлоогүй бол compiler автоматаар хоосон байгуулагчийг үүсгэнэ.

Мөн класст нэгээс олон байгуулагч функц тодорхойлж өгч болно. Энэ нь дахин тодорхойлогдож байна гэсэн үг болно. Хэд хэдэн байгуулагч функцтай класс зарлах үед тэдгээр байгуулагч функцууд нь хоорондоо параметрийнхээ төрлөөр ялгагдаж байх ёстой. Ингээгүй тохиолдолд compiler байгуулагч функцуудын алийг нь дуудахаа тодорхойлж чадахгүй байдалд хүрч алдаа өгнө.

классын\_нэр() {

…

}

классын\_нэр(төрөл параметрын\_нэр, ...) {

...

}

## 3.2 Устгагч функц

Устгагч функц нь обьектын динамик өгөгдлийн санах ойг чөлөөлөхөд хэрэглэгдэнэ. Класс ганц устгагч функцтэй байна. Хэрэв класст устгагч функц байхгүй бол compiler обьектийн өгөгдлийн нөөцөлсөн санах ойг автоматаар чөлөөлнө. Устгагч функцыг дуудаж ажиллуулдаггүй бөгөөд үйлчлэх хүрээ дуусах үед compiler автоматаар дуудаж ажиллуулна. Утга буцаахгүй бөгөөд параметр авахгүй.

~классын\_нэр() {

…

}

## 3.3 Хуулагч байгуулагч

Хуулагч байгуулагч функцын тусламжтайгаар аль хэдийн үүссэн объектын гишүүн өгөгдлүүдтэй ижил гишүүн өгөгдөлтэй объектыг байгуулна.

классын\_нэр (классын\_нэр &объектын\_нэр) {

…

}

объектын\_нэр = объектын\_нэр;

Дээрх үйлдлийн тусламжтайгаар эхний объектод дараагийн объектын бүх гишүүн өгөгдлийг хуулж тавина. Гэхдээ класс дотор хаяган хувьсагч байгаа тохиолдолд дээрх үйлдэл

тохиромжгүй юм. Учир нь нэг санах ойд хоёр объект хоёул хандаж эхэлнэ. Мөн санах ойн

цоорхойг үүсгэнэ. Санах ойн цоорхойгоос сэргийлэхийн тулд динамикаар нөөцөлсөн санах

ойг хэрэглэж дууссаны дараа устгах хэрэгтэй.

## 3.4 Хуулагч функц

Хуулагч функц нь үүссэн объектын утгыг өөр объектод хуулан хоёр объектыг ижил болгох зорилготой функц юм.

Хуулагч функц болон хуулагч байгуулагчийн ялгаа нь хуулагч байгуулагч нь шинэ объект үүсэхдээ л аль нэг объектын гишүүн өгөгдлөөр утгаа авдаг бол хуулагч функц нь объект үүссэн л бол хэзээ ч өөр объектын утгыг авах боломжтой байдаг. Мөн хуулагч байгуулагч объектын хувьд хамгийн ихдээ нэг л дуудагдаж ажиллах бол хуулагч функц нь хэдэн ч удаа дуудагдах боломжтой.

# 4. ХЭРЭГЖҮҮЛЭЛТ

Хамгийн түрүүнд классыг тодорхойлсон.

class Ajilchin {

private:

    int dugaar;

    float ajillasan\_tsag;

    char\* ner;

    char\* alban\_tushaal;

    float zahirliin\_tsalin\_bodoh();

public:

    Ajilchin();

    ~Ajilchin();

    Ajilchin(Ajilchin&);

    Ajilchin(int,char\*,char\*,float);

    void set\_dugaar(int);

    void set\_ner(char\*);

    void set\_alban\_tushaal(char\*);

    void set\_ajillasan\_tsag(float);

    bool ajillasan\_tsag\_nemegduuleh(float);

    int get\_dugaar();

    char\* get\_ner();

    char\* get\_alban\_tushaal();

    float get\_ajillasan\_tsag();

    float tsalin\_bodoh();

};

Түүний дараа гишүүн функцуудыг тодорхойлсон.

Тэгээд нэрээр эрэмбэлдэг, дугаар давхацсан эсэхийг шалгадаг мөн хуулагч хэрэглэгчийн функцыг тодорхойлсон.

int dugaar\_davhtssan\_eseh(Ajilchin a[], int n, int id) {

    int k=0;

    for(int i=0; i<n; i++) {

        if(a[i].get\_dugaar() == id) {

            k++;

        }

    }

    if(k==0)

        return 0;

    return 1;

}

void nereer\_erembeleh(Ajilchin a[], int n) {

    int sav1, sav2;

    char sav3[20], sav4[20];

    float sav5, sav6;

    for(int j=0; j<=n; j++) {

        for(int i=0; i<n; i++) {

            if(strcmp(a[i].get\_ner(), a[i+1].get\_ner())>0) {

                sav1 = a[i].get\_dugaar();

                sav2 = a[i+1].get\_dugaar();

                a[i].set\_dugaar(sav2);

                a[i+1].set\_dugaar(sav1);

                strcpy(sav3, a[i].get\_ner());

                strcpy(sav4, a[i+1].get\_ner());

                a[i].set\_ner(sav4);

                a[i+1].set\_ner(sav3);

                strcpy(sav3, a[i].get\_alban\_tushaal());

                strcpy(sav4, a[i+1].get\_alban\_tushaal());

                a[i].set\_alban\_tushaal(sav4);

                a[i+1].set\_alban\_tushaal(sav3);

                sav5 = a[i].get\_ajillasan\_tsag();

                sav6 = a[i+1].get\_ajillasan\_tsag();

                a[i].set\_ajillasan\_tsag(sav6);

                a[i+1].set\_ajillasan\_tsag(sav5);

            }

        }

    }

}

void ajilchin\_huulah(Ajilchin &a, Ajilchin &b) {

    int sav1;

    char sav2[50];

    float sav3;

    sav1 = b.get\_dugaar();

    a.set\_dugaar(sav1);

    strcpy(sav2, b.get\_ner());

    a.set\_ner(sav2);

    strcpy(sav2, b.get\_alban\_tushaal());

    a.set\_alban\_tushaal(sav2);

    sav3 = b.get\_ajillasan\_tsag();

    a.set\_ajillasan\_tsag(sav3);

}

Main функц дотор уг классын объектон хүснэгт үүсгэсэн бөгөөд Main функц нь давталтаар зохион байгуулагдсан бөгөөд гарах нөхцлийг хангах хүртэл програм ажиллаж ажилчин нэмэх, мэдээллийг оруулах, мэдээллийг харах, нэрсээр нь эрэмбэлэх үйлдлүүдийг хийнэ.

main() {

    int m=0, n=-1;

    int a;

    char b[20], c[20];

    float d;

    Ajilchin k[30];

    while(m != 7) {

        cout << "1. Ajiltan nemeh\n";

        cout << "2. Ajiltnii medeelliig oruulah\n";

        cout << "3. Ajilchdiin medeelliig delgetsleh\n";

        cout << "4. Ajilchdiig nereer ni erembeleh\n";

        cout << "5. Ajillasan tsag nemegduuleh\n";

        cout << "6. Ajilchnii medeelliig huulah\n";

        cout << "7. Duusgah\n";

        cin >> m;

        if(m==1) {

            n++;

        }

        if(m==2) {

            cout << "dugaar oruul: ";

            cin >> a;

            if(dugaar\_davhtssan\_eseh(k, n, a)==0) {

                k[n].set\_dugaar(a);

                cout << "ner oruul: ";

                cin >> b;

                k[n].set\_ner(b);

                cout << "alban tushaal oruul: ";

                cin >> c;

                k[n].set\_alban\_tushaal(c);

            } else {

                cout << "Iim id dugaartai ajiltan burtgeltei baina.\n";

            }

        }

        if(m==3) {

            for(int i=0; i<=n; i++) {

                cout << k[i].get\_dugaar() << " dugaartai ajiltan ";

                cout << k[i].get\_ner() << " ni ";

                cout << k[i].get\_alban\_tushaal() << " alban tushaaltai buguud ";

                cout << k[i].get\_ajillasan\_tsag() << " tsag ajillasan ba ";

                cout << k[i].tsalin\_bodoh() << " tsalintai baina.\n";

            }

        }

        if(m==4) {

            nereer\_erembeleh(k, n);

        }

        if(m==5) {

            int h=0;

            cout << "Tsagiig ni nemeh ajiltnii dugaariig oruul: ";

            cin >> a;

            for(int i=0; i<=n; i++) {

                if(k[i].get\_dugaar()==a) {

                    h++;

                }

            }

            for(int i=0; i<=n; i++) {

                if(k[i].get\_dugaar()==a) {

                    cout << "Nemeh tsagaa oruul: ";

                    cin >> d;

                    if(k[i].ajillasan\_tsag\_nemegduuleh(d)==1) {

                        cout << "Ajiltnii tsag amjilttai nemegdlee.\n";

                    } else {

                        cout << "Buruu tsag oruulsan baina.\n";

                    }

                }

            }

            if(h==0) {

                cout << "Tanii oruulsan dugaartai ajiltan baihgui baina\n";

            }

        }

        if(m==6) {

            int haih;

            cout << "medeelliig huulah ajiltnii dugaariig oruul: ";

            cin >> haih;

            for(int i=0; i<n; i++) {

                if(k[i].get\_dugaar()==haih) {

                    ajilchin\_huulah(k[n], k[i]);

                }

            }

        }

    }

}

Ажилтны цалинг ажилласан цаг дээр үндэслэн бодож ажилчдын мэдээллийг дэлгэцлэхэд харагдана. Захиралын цалинг цагийн 10000, бусад ажилчдын цалинг цагийн 5000 гэж үзсэн.

float Ajilchin::tsalin\_bodoh() {

    if(strcmp(alban\_tushaal, "zahiral")==0) {

        return zahirliin\_tsalin\_bodoh();

    }

    int tsagiin\_huls = 5000;

    return tsagiin\_huls \* ajillasan\_tsag;

}

float Ajilchin::zahirliin\_tsalin\_bodoh() {

    int tsagiin\_huls = 10000;

    return tsagiin\_huls \* ajillasan\_tsag;

}

Main функц дуусах үед объектын устгагч функц автоматаар дуудагдаж нөөцөлсөн санах ойг устгаад устлаа гэдэг үгийг дэлгэцлэнэ.

# 5. ДҮГНЭЛТ

Байгуулагч функцийг тодорхойлж өгснөөр классын объект үүсэх үед гарааны утгыг шууд оноох боломжтой болгож байна. Динамик байдлаар санах ойг нөөцөлж ашиглах нь санах ойн замбараагүй хэрэглээг багасгах сайн талтай хэдий ч хэрэв нөөцөлсөн санах ойгоо устгахаа мартвал санах ойн цоорхой үүсэх эрсдэлтэй.

# 6. АШИГЛАСАН МАТЕРИАЛ

1. https://www.geeksforgeeks.org/

# 7. ХАВСРАЛТ

#include <iostream>

#include <string.h>

using namespace std;

class Ajilchin {

private:

    int dugaar;

    float ajillasan\_tsag;

    char\* ner;

    char\* alban\_tushaal;

    float zahirliin\_tsalin\_bodoh();

public:

    Ajilchin();

    ~Ajilchin();

    Ajilchin(Ajilchin&);

    Ajilchin(int,char\*,char\*,float);

    void set\_dugaar(int);

    void set\_ner(char\*);

    void set\_alban\_tushaal(char\*);

    void set\_ajillasan\_tsag(float);

    bool ajillasan\_tsag\_nemegduuleh(float);

    int get\_dugaar();

    char\* get\_ner();

    char\* get\_alban\_tushaal();

    float get\_ajillasan\_tsag();

    float tsalin\_bodoh();

};

int dugaar\_davhtssan\_eseh(Ajilchin a[], int n, int id) {

    int k=0;

    for(int i=0; i<n; i++) {

        if(a[i].get\_dugaar() == id) {

            k++;

        }

    }

    if(k==0)

        return 0;

    return 1;

}

void nereer\_erembeleh(Ajilchin a[], int n) {

    int sav1, sav2;

    char sav3[20], sav4[20];

    float sav5, sav6;

    for(int j=0; j<=n; j++) {

        for(int i=0; i<n; i++) {

            if(strcmp(a[i].get\_ner(), a[i+1].get\_ner())>0) {

                sav1 = a[i].get\_dugaar();

                sav2 = a[i+1].get\_dugaar();

                a[i].set\_dugaar(sav2);

                a[i+1].set\_dugaar(sav1);

                strcpy(sav3, a[i].get\_ner());

                strcpy(sav4, a[i+1].get\_ner());

                a[i].set\_ner(sav4);

                a[i+1].set\_ner(sav3);

                strcpy(sav3, a[i].get\_alban\_tushaal());

                strcpy(sav4, a[i+1].get\_alban\_tushaal());

                a[i].set\_alban\_tushaal(sav4);

                a[i+1].set\_alban\_tushaal(sav3);

                sav5 = a[i].get\_ajillasan\_tsag();

                sav6 = a[i+1].get\_ajillasan\_tsag();

                a[i].set\_ajillasan\_tsag(sav6);

                a[i+1].set\_ajillasan\_tsag(sav5);

            }

        }

    }

}

void ajilchin\_huulah(Ajilchin &a, Ajilchin &b) {

    int sav1;

    char sav2[50];

    float sav3;

    sav1 = b.get\_dugaar();

    a.set\_dugaar(sav1);

    strcpy(sav2, b.get\_ner());

    a.set\_ner(sav2);

    strcpy(sav2, b.get\_alban\_tushaal());

    a.set\_alban\_tushaal(sav2);

    sav3 = b.get\_ajillasan\_tsag();

    a.set\_ajillasan\_tsag(sav3);

}

Ajilchin::Ajilchin() {

    dugaar = 0;

    ner = new char[4];

    strcpy(ner, "bat");

    alban\_tushaal = new char[9];

    strcpy(alban\_tushaal, "ajilchin");

    ajillasan\_tsag = 0;

}

Ajilchin::~Ajilchin() {

    delete ner;

    delete alban\_tushaal;

    cout << "ustlaa.\n";

}

Ajilchin::Ajilchin(Ajilchin& staff) {

    dugaar = staff.dugaar;

    delete ner;

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

    strcpy(ner, staff.ner);

    delete alban\_tushaal;

    alban\_tushaal = new char [strlen(staff.alban\_tushaal)+1];

    strcpy(alban\_tushaal, staff.alban\_tushaal);

    ajillasan\_tsag = staff.ajillasan\_tsag;

}

Ajilchin::Ajilchin(int a, char\* b, char\* c, float d) {

    dugaar = a;

    delete ner;

    ner = new char[strlen(b)+1];

    strcpy(ner, b);

    delete alban\_tushaal;

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

    strcpy(alban\_tushaal, c);

    ajillasan\_tsag = d;

}

void Ajilchin::set\_dugaar(int a) {

    dugaar = a;

}

void Ajilchin::set\_ner(char\* a) {

    delete ner;

    ner = new char[strlen(a)+1];

    strcpy(ner, a);

}

void Ajilchin::set\_alban\_tushaal(char\* a) {

    delete alban\_tushaal;

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

    strcpy(alban\_tushaal, a);

}

void Ajilchin::set\_ajillasan\_tsag(float a) {

    ajillasan\_tsag = a;

}

bool Ajilchin::ajillasan\_tsag\_nemegduuleh(float a) {

    if(a>=0 && a<=24) {

        ajillasan\_tsag += a;

        return true;

    }

    return false;

}

int Ajilchin::get\_dugaar() {

    return dugaar;

}

char\* Ajilchin::get\_ner() {

    return ner;

}

char\* Ajilchin::get\_alban\_tushaal() {

    return alban\_tushaal;

}

float Ajilchin::get\_ajillasan\_tsag() {

    return ajillasan\_tsag;

}

float Ajilchin::tsalin\_bodoh() {

    if(strcmp(alban\_tushaal, "zahiral")==0) {

        return zahirliin\_tsalin\_bodoh();

    }

    int tsagiin\_huls = 5000;

    return tsagiin\_huls \* ajillasan\_tsag;

}

float Ajilchin::zahirliin\_tsalin\_bodoh() {

    int tsagiin\_huls = 10000;

    return tsagiin\_huls \* ajillasan\_tsag;

}

main() {

    int m=0, n=-1;

    int a;

    char b[20], c[20];

    float d;

    Ajilchin k[30];

    while(m != 7) {

        cout << "1. Ajiltan nemeh\n";

        cout << "2. Ajiltnii medeelliig oruulah\n";

        cout << "3. Ajilchdiin medeelliig delgetsleh\n";

        cout << "4. Ajilchdiig nereer ni erembeleh\n";

        cout << "5. Ajillasan tsag nemegduuleh\n";

        cout << "6. Ajilchnii medeelliig huulah\n";

        cout << "7. Duusgah\n";

        cin >> m;

        if(m==1) {

            n++;

        }

        if(m==2) {

            cout << "dugaar oruul: ";

            cin >> a;

            if(dugaar\_davhtssan\_eseh(k, n, a)==0) {

                k[n].set\_dugaar(a);

                cout << "ner oruul: ";

                cin >> b;

                k[n].set\_ner(b);

                cout << "alban tushaal oruul: ";

                cin >> c;

                k[n].set\_alban\_tushaal(c);

            } else {

                cout << "Iim id dugaartai ajiltan burtgeltei baina.\n";

            }

        }

        if(m==3) {

            for(int i=0; i<=n; i++) {

                cout << k[i].get\_dugaar() << " dugaartai ajiltan ";

                cout << k[i].get\_ner() << " ni ";

                cout << k[i].get\_alban\_tushaal() << " alban tushaaltai buguud ";

                cout << k[i].get\_ajillasan\_tsag() << " tsag ajillasan ba ";

                cout << k[i].tsalin\_bodoh() << " tsalintai baina.\n";

            }

        }

        if(m==4) {

            nereer\_erembeleh(k, n);

        }

        if(m==5) {

            int h=0;

            cout << "Tsagiig ni nemeh ajiltnii dugaariig oruul: ";

            cin >> a;

            for(int i=0; i<=n; i++) {

                if(k[i].get\_dugaar()==a) {

                    h++;

                }

            }

            for(int i=0; i<=n; i++) {

                if(k[i].get\_dugaar()==a) {

                    cout << "Nemeh tsagaa oruul: ";

                    cin >> d;

                    if(k[i].ajillasan\_tsag\_nemegduuleh(d)==1) {

                        cout << "Ajiltnii tsag amjilttai nemegdlee.\n";

                    } else {

                        cout << "Buruu tsag oruulsan baina.\n";

                    }

                }

            }

            if(h==0) {

                cout << "Tanii oruulsan dugaartai ajiltan baihgui baina\n";

            }

        }

        if(m==6) {

            int haih;

            cout << "medeelliig huulah ajiltnii dugaariig oruul: ";

            cin >> haih;

            for(int i=0; i<n; i++) {

                if(k[i].get\_dugaar()==haih) {

                    ajilchin\_huulah(k[n], k[i]);

                }

            }

        }

    }

}

# Оператор Объектод Дахин Тодорхойлох Лабораторийн Ажил (Лаборатори №6)

# 1. ОРШИЛ/УДИРТГАЛ

Матриц классд операторыг дахин тодорхойлж тухайн операторын тусламжтайгаар матриц дээр матрицыг нэмэх, хасах, үржих, тоог нэмэх, хасах, үржих зэрэг үйлдлүүдийг хийсэн.

# 2. ЗОРИЛГО

Матриц классыг өгөгдсөн шаардлагын дагуу зарлаж, гишүүн функцуудийг тодорхойлж, янз бүрийн боловсруулалт хийсэн. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Гишүүн өгөгдлүүдийн төрлийг тодорхойлсон,
2. Параметертэй байгуулагч функц тодорхойлсон,
3. Гишүүн функцүүдийг тодорхойлсон,
4. Операторуудыг дахин тодорхойлсон.

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 Дахин тодорхойлогдсон оператор

Энгийн операторыг объектын илэрхийлэлд ашиглахдаа тодорхой боловсруулах дүрэм зааж болдог. Үүнийг дахин тодорхойлогдсон оператор гэнэ.

Оператор дахин тодорхойлохдоо:

буцах\_утгын\_төрөл **operator** оператор (параметрүүд) {

…

}

Жишээ нь:

void Counter::operator ++() {

count++;

}

гэж тодорхойлж болно. Оператороо ашиглахдаа дараах байдлаар ашиглана:

Counter c1;

c1++;

эсвэл

c1.operator ++();

гэж ашиглана.

Оператор дахин тодорхойлохдоо шинэ оператор үүсгэж тодорхойлж болохгүй. Мөн

дахин тодорхойлогдсон оператор эх операторын дүрмийг зөрчихгүй.

. \* :: ?: sizeof() операторуудыг дахин тодорхойлохгүй.

# 4. ХЭРЭГЖҮҮЛЭЛТ

Матрицын мөр баганын тоог илэрхийлэх хоёр бүхэл тоо болон 2хэмжээст хүснэгтийг агуулах боломжтой хаяган хувьсагчийг класст тодорхойлсон. Үүний дараа байгуулагч болон устгагч функц, set функцыг тодорхойлсон. Операторуудыг мөн дахин тодорхойлсон.

class Matrix {

private:

    int n, m;

    float \*\*values;

public:

    Matrix();

    Matrix(int n, int m=1);

    ~Matrix();

    void set\_matrix(float a[100][100], int, int);

    Matrix operator +(float);

    Matrix operator +(Matrix);

    Matrix operator \*(Matrix);

    Matrix operator -(Matrix);

    void operator =(Matrix);

    void operator ++();

    void operator --();

    void operator +=(float);

    void operator -=(float);

    void operator \*=(float);

    float get\_num(int, int);

    void operator /=(int);

    void show();

};

эдгээр операторуудаас хамгийн ээдрээтэй байсан нь үржих оператор байв. Учир нь матриц үржүүлэх дүрэм тоо үржүүлэхээс тэс өөр. Үржих операторыг дараах байдлаар тодорхойлов.

Matrix Matrix::operator \*(Matrix a) {

    if(m != a.n) {

        cout << "Ug hoyr matritsiig urjih bolomjgui!" << endl;

        return 0;

    }

    float k[100][100];

    for(int i=0; i<n; i++) {

        for(int j=0; j<a.m; j++) {

            k[i][j] = 0;

            for(int l=0; l<m; l++) {

                k[i][j] += values[i][l] \* a.values[l][j];

            }

        }

    }

    Matrix matrits(n, a.m);

    matrits.set\_matrix(k, n, a.m);

    return matrits;

}

Майн функц нь давталтаар зохион байгуулагдсан бөгөөд давталт дуусах нөхцөл хангагдтал матриц дээрх үйлдүүдийг хийх боломжтой байдлаар зохион байгуулсан.

main() {

    int a;

    float u[100][100];

    int n, m;

    while(true) {

        cout << "1.matrits deer matrits nemeh\n";

        cout << "2.matrits deer too nemeh\n";

        cout << "3.matritsiig matritsaar urjih\n";

        cout << "4.matritsaas matrits hasah\n";

        cout << "5.matritsiin utgiig negeer nemegduuleh\n";

        cout << "6.matritsiin utgiig negeer buuruulah\n";

        cout << "7.matritsiin utgiig nemegduuleh\n";

        cout << "8.matritsiin utgiig buuruulah\n";

        cout << "9.matritsiin utgiig urjih\n";

        cout << "10.matritsaas element harah\n";

        cout << "11.matrits horvuuleh\n";

        cout << "12.duusgah\n";

        cin >> a;

        if(a==1) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            Matrix matrits2;

            cout << "Nemeh matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits2.set\_matrix(u,n,m);

            Matrix matrits3 = matrits1 + matrits2;

            cout << "hariu:\n";

            matrits3.show();

        } if(a==2) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            cout << "Nemeh toogoo oruul: ";

            float too;

            cin >> too;

            Matrix matrits2 = matrits1 + too;

            matrits2.show();

        } if(a==3) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            Matrix matrits2;

            cout << "Urjih matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits2.set\_matrix(u,n,m);

            Matrix matrits3 = matrits1 \* matrits2;

            matrits3.show();

        } if(a==4) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            Matrix matrits2;

            cout << "Hasah matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits2.set\_matrix(u,n,m);

            Matrix matrits3 = matrits1 - matrits2;

            matrits3.show();

        } if(a==5) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            ++matrits1;

            matrits1.show();

        } if(a==6) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            --matrits1;

            matrits1.show();

        } if(a==7) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            float too;

            cout << "nemeh toogoo oruul: ";

            cin >> too;

            matrits1 += too;

            matrits1.show();

        } if(a==8) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            float too;

            cout << "Hasah toogoo oruul: ";

            cin >> too;

            matrits1 -= too;

            matrits1.show();

        } if(a==9) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            float too;

            cout << "urjih toogoo oruul: ";

            cin >> too;

            matrits1 \*= too;

            matrits1.show();

        } if(a==10) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            cout << "Haih elementiin mor baganiin dugaariig oruul: ";

            int mor, bagana;

            cin >> mor >> bagana;

            cout << matrits1.get\_num(mor-1, bagana-1) << endl;

        } if(a==11) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            matrits1/=1;

            matrits1.show();

        } if(a==12) {

            break;

        }

    }

}

# 5. ДҮГНЭЛТ

Хэрэв класст операторыг дахин тодорхойлохгүй бол объект дээр боловсруулалт хийхийн тулд шаардалагатай өгөгдлүүдийг цуглуулж, боловсруулан утгыг өөрчлөх гишүүдийн утгыг өөрчлөнө. Харин операторыг дахин тодорхойлсноор класс дотроо боловсруулалт хийн утгыг буцааж тодорхой дүрмийн дагуу боловсруулалт хийх боломжийг олгож байна.

# 6. АШИГЛАСАН МАТЕРИАЛ

1. <https://www.geeksforgeeks.org/operator-overloading-c/>

# 7. ХАВСРАЛТ

#include <iostream>

using namespace std;

class Matrix {

private:

    int n, m;

    float \*\*values;

public:

    Matrix();

    Matrix(int n, int m=1);

    ~Matrix();

    void set\_matrix(float a[100][100], int, int);

    Matrix operator +(float);

    Matrix operator +(Matrix);

    Matrix operator \*(Matrix);

    Matrix operator -(Matrix);

    void operator =(Matrix);

    void operator ++();

    void operator --();

    void operator +=(float);

    void operator -=(float);

    void operator \*=(float);

    float get\_num(int, int);

    void operator /=(int);

    void show();

};

Matrix::Matrix() {

    n=1; m=1;

    values = new float \*[n];

    for(int i=0; i<n; i++) {

        values[i] = new float [m];

    }

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] = 0;

        }

    }

}

Matrix::Matrix(int n, int m) {

    this->n = n; this->m = m;

    values = new float \*[n];

    for(int i=0; i<n; i++) {

        values[i] = new float [m];

    }

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] = 0;

        }

    }

}

Matrix::~Matrix() {

    for(int i=0; i<n; i++) {

        delete values[i];

    }

    delete values;

}

void Matrix::set\_matrix(float a[100][100], int mor, int bagana) {

    for(int i=0; i<n; i++) {

        delete values[i];

    }

    delete values;

    values = new float \*[mor];

    for(int i=0; i<mor; i++) {

        values[i] = new float [bagana];

    }

    n = mor;   m = bagana;

    for(int i=0; i<mor; i++) {

        for(int j=0; j<bagana; j++) {

            values[i][j] = a[i][j];

        }

    }

}

Matrix Matrix::operator +(float a) {

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] += a;

        }

    }

    return \*this;

}

Matrix Matrix::operator +(Matrix a) {

    if(n!=a.n || m!=a.m) {

        cout << "Ug hoyr matritsiig nemeh bolomjgui!" << endl;

        return \*this;

    }

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] += a.values[i][j];

        }

    }

    return \*this;

}

Matrix Matrix::operator \*(Matrix a) {

    if(m != a.n) {

        cout << "Ug hoyr matritsiig urjih bolomjgui!" << endl;

        return 0;

    }

    float k[100][100];

    for(int i=0; i<n; i++) {

        for(int j=0; j<a.m; j++) {

            k[i][j] = 0;

            for(int l=0; l<m; l++) {

                k[i][j] += values[i][l] \* a.values[l][j];

            }

        }

    }

    Matrix matrits(n, a.m);

    matrits.set\_matrix(k, n, a.m);

    return matrits;

}

Matrix Matrix::operator -(Matrix a) {

    if(n!=a.n || m!=a.m) {

        cout << "Ug hoyr matritsiig hasah bolomjgui!" << endl;

        return \*this;

    }

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] -= a.values[i][j];

        }

    }

    return \*this;

}

void Matrix::operator =(Matrix a) {

    for(int i=0; i<n; i++) {

        delete values[i];

    }

    delete values;

    values = new float \*[a.n];

    for(int i=0; i<n; i++) {

        values[i] = new float [a.m];

    }

    this->n = a.n;  this->m = a.m;

    for(int i=0; i<a.n; i++) {

        for(int j=0; j<a.m; j++) {

            values[i][j] = a.values[i][j];

        }

    }

}

void Matrix::operator ++() {

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j]++;

        }

    }

}

void Matrix::operator --() {

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j]--;

        }

    }

}

void Matrix::operator +=(float a) {

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] += a;

        }

    }

}

void Matrix::operator -=(float a) {

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] -= a;

        }

    }

}

void Matrix::operator \*=(float a) {

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            values[i][j] \*= a;

        }

    }

}

float Matrix::get\_num(int a, int b) {

    return values[a][b];

}

void Matrix::operator /=(int k) {

    float a[m][n];

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            a[j][i] = values[i][j];

        }

    }

    for(int i=0; i<n; i++) {

        delete values[i];

    }

    delete values;

    values = new float \*[m];

    for(int i=0; i<n; i++) {

        values[i] = new float [n];

    }

    m = m + n;

    n = m - n;

    m = m - n;

    for(int i=0; i<m; i++) {

        for(int j=0; j<n; j++) {

            values[j][i] = a[j][i];

        }

    }

}

void Matrix::show() {

    for(int i=0; i<n; i++) {

        for(int j=0; j<m; j++) {

            cout << values[i][j] << "\t";

        }

    cout << endl;

    }

}

main() {

    int a;

    float u[100][100];

    int n, m;

    while(true) {

        cout << "1.matrits deer matrits nemeh\n";

        cout << "2.matrits deer too nemeh\n";

        cout << "3.matritsiig matritsaar urjih\n";

        cout << "4.matritsaas matrits hasah\n";

        cout << "5.matritsiin utgiig negeer nemegduuleh\n";

        cout << "6.matritsiin utgiig negeer buuruulah\n";

        cout << "7.matritsiin utgiig nemegduuleh\n";

        cout << "8.matritsiin utgiig buuruulah\n";

        cout << "9.matritsiin utgiig urjih\n";

        cout << "10.matritsaas element harah\n";

        cout << "11.matrits horvuuleh\n";

        cout << "12.duusgah\n";

        cin >> a;

        if(a==1) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            Matrix matrits2;

            cout << "Nemeh matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits2.set\_matrix(u,n,m);

            Matrix matrits3 = matrits1 + matrits2;

            cout << "hariu:\n";

            matrits3.show();

        } if(a==2) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            cout << "Nemeh toogoo oruul: ";

            float too;

            cin >> too;

            Matrix matrits2 = matrits1 + too;

            matrits2.show();

        } if(a==3) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            Matrix matrits2;

            cout << "Urjih matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits2.set\_matrix(u,n,m);

            Matrix matrits3 = matrits1 \* matrits2;

            matrits3.show();

        } if(a==4) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            Matrix matrits2;

            cout << "Hasah matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits2.set\_matrix(u,n,m);

            Matrix matrits3 = matrits1 - matrits2;

            matrits3.show();

        } if(a==5) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            ++matrits1;

            matrits1.show();

        } if(a==6) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            --matrits1;

            matrits1.show();

        } if(a==7) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            float too;

            cout << "nemeh toogoo oruul: ";

            cin >> too;

            matrits1 += too;

            matrits1.show();

        } if(a==8) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            float too;

            cout << "Hasah toogoo oruul: ";

            cin >> too;

            matrits1 -= too;

            matrits1.show();

        } if(a==9) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            float too;

            cout << "urjih toogoo oruul: ";

            cin >> too;

            matrits1 \*= too;

            matrits1.show();

        } if(a==10) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            cout << "Haih elementiin mor baganiin dugaariig oruul: ";

            int mor, bagana;

            cin >> mor >> bagana;

            cout << matrits1.get\_num(mor-1, bagana-1) << endl;

        } if(a==11) {

            Matrix matrits1;

            cout << "Ehnii matritsiin mor baganiin toog oruul: ";

            cin >> n >> m;

            cout << "Elementuudiig oruul: ";

            for(int i=0; i<n; i++) {

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

                    cin >> u[i][j];

                }

            }

            matrits1.set\_matrix(u,n,m);

            matrits1/=1;

            matrits1.show();

        } if(a==12) {

            break;

        }

    }

}

#include <iostream>

#include <string.h>

#include <math.h>

using namespace std;

class Shape {

public:

    char ner[20];

};

class TwoDShape:public Shape {

public:

    int oroin\_too;

};

class Circle:public TwoDShape {

public:

    float toviin\_koordinat[1];

    float radius;

    Circle(float a[1], float r) {

        toviin\_koordinat[0] = a[0];

        toviin\_koordinat[1] = a[1];

        radius = r;

    };

    float toirgiin\_talbai() {

        return M\_PI\*radius\*radius;

    }

    float toirgiin\_urt() {

        return 2\*M\_PI\*radius;

    }

};

class Square:public TwoDShape {

public:

    float zuun\_deed\_oroi[1];

    float baruun\_deed\_oroi[1];

    float zuun\_dood\_oroi[1];

    float baruun\_dood\_oroi[1];

    float taliin\_urt;

    Square(float a[1], float d) {

        zuun\_deed\_oroi[0] = a[0];

        zuun\_deed\_oroi[1] = a[1];

        zuun\_dood\_oroi[0] = a[0];

        zuun\_dood\_oroi[1] = a[1] - d;

        baruun\_deed\_oroi[0] = a[0] + d;

        baruun\_deed\_oroi[1] = a[1];

        baruun\_dood\_oroi[0] = a[0] + d;

        baruun\_dood\_oroi[1] = a[1] - d;

        taliin\_urt = d;

    }

    float kv\_talbai() {

        return taliin\_urt\*taliin\_urt;

    }

    float kv\_pirimetr() {

        return 4\*taliin\_urt;

    }

};

class Triangle:public TwoDShape {

public:

    float oroin\_tseg[1];

    float suuri\_zuun[1];

    float suuri\_baruun[1];

    float d;

    Triangle(float a[1], float k) {

        d = k;

        oroin\_tseg[0] = a[0];

        oroin\_tseg[1] = a[1];

        suuri\_zuun[0] = a[0] - d/2;

        suuri\_zuun[1] = a[1] - sin(M\_PI/3)\*d;

        suuri\_baruun[0] = a[0] + d/2;

        suuri\_baruun[1] = a[1] -sin(M\_PI/3)\*d;

    }

    float gurvaljnii\_talbai() {

        return d\*d\*sin(M\_PI\*60/180)/2;

    }

    float gurvaljnii\_pirimetr() {

        return d\*3;

    }

};

main() {

    float a[1], b;

    cout << "toirgiin toviin koordinat bolon radiusiig oruul: ";

    cin >> a[0] >> a[1] >> b;

    Circle toirog(a, b);

    cout << toirog.toirgiin\_talbai() << endl;

    cout << toirog.toirgiin\_urt() << endl;

    cout << "Kvadratiin zuun deed tsegiin koordinat bolon taliin urtiig oruul: ";

    cin >> a[0] >> a[1] >> b;

    Square kvadrat(a, b);

    cout << kvadrat.kv\_talbai() << endl;

    cout << kvadrat.kv\_pirimetr() << endl;

    cout << "gurvaljnii oroin koordinat bolon taliin urtiig oruul: ";

    cin >> a[0] >> a[1] >> b;

    Triangle gurvaljin(a, b);

    cout << gurvaljin.gurvaljnii\_talbai() << endl;

    cout << gurvaljin.gurvaljnii\_pirimetr() << endl;

}