

Лаборатори 6:

1. Оператор дахин тодорхойлох гэж юу вэ? Хэрхэн тодорхойлдог вэ?

Оператор дахин тодорхойлох гэдэг нь энгийн операторыг объектын илэрхийлэлд ашиглахаар тодорхойлохыг хэлнэ. С++ хэлэнд шинээр оператор үүсгэж болохгүй. Харин операторыг дахин тодорхойлж болдог. Дахин тодорхойлогдох оператор эх операторын дүрмийг зөрчиж болохгүй.

(+ - * / = <> += -= *= /= << >> == != <= >= ++ -- & ! && // [] () new delete)
дахин тодорхойлох боломжтой.

```
returnType operator temdegt (argument) {
```

```
...
```

```
}
```

```
class Matrix {
private:
    int m;          // muriin too
    int n;          // baganiin too
    float *values;  // matrixiin utguud
public:
    // Mur болон bagand anhi utga olgoh anhdagch baiguulagch;
    Matrix(int m = 1, int n = 1)
    {
        m = m;
        n = n;
        values = new float[m * n];
    };
    // Ustgagch;
    ~Matrix()
    {
        delete[] values;
    }
    // Mur ба baganiin indexdeh matrixiin utgiig butsaah;
```

```

float &at(int r, int c)
{
    return values[n * r + c];
}
// Baginiin too uurchluh;
void setCol(int col)
{
    n = col;
    values = new float[m * n];
}
// Muriin too uurchluh;
void setRow(int row)
{
    m = row;
    values = new float[m * n];
}
// Mur ba baganiin indexdeh matrixiin utgiig uurchluh;
void setValues(int r, int c, float num)
{
    at(r, c) = num;
}
// Baganiin too butsaah;
int getCol()
{
    return n;
}
// Muriin too butsaah;
int getRow()
{
    return m;
}
// Mur ba baganiin indexdeh matrixiin utgiig butsaah;
float getValues(int r, int c)
{
    return at(r, c);
}

```

```

}
// Matrixiin utgiig garaas avah;
void inputValues()
{
    float num;
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            cout << "[" << i << "]"
                << "[" << j << "]: ";
            cin >> num;
            setValues(i, j, num);
        }
    }
}

void printValues()
{
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            cout << getValues(i, j) << " ";
        }
        cout << "\n";
    }
}

// Matrix deer bodit too nemeed niilberiig butsaana;
Matrix operator+(float num)
{
    Matrix res(m, n);

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

```

```

        {
            int sum = getValues(i, j) + num;
            res.setValues(i, j, sum);
        }
    }
    return res;
}

// 2 Matrixiig nemeed niilberiig butsaah;
Matrix operator+(Matrix &mat)
{
    Matrix res(m, n);

    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int sum = getValues(i, j) + mat.getValues(i, j);
            res.setValues(i, j, sum);
        }
    }
    return res;
}

// 2 Matrixiig urjij urjveriig butsaah;
Matrix operator*(Matrix &mat)
{
    Matrix tmp(m, mat.n);
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < mat.n; j++)
        {
            int res = 0;
            for (int k = 0; k < mat.m; k++)
            {
                res += getValues(i, k) * mat.getValues(k, j);
            }
        }
    }
}

```

```

        }
        tmp.setValues(i, j, res);
    }
}
return tmp;
}
// 2 Matrixiig hasaad ylgavriig butsaah;
Matrix operator-(Matrix &mat)
{
    Matrix res(m, n);
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int sub = getValues(i, j) - mat.getValues(i, j);
            res.setValues(i, j, sub);
        }
    }
    return res;
}
// 1 matrixiin objectiig nuguud utga onooh;
void operator=(Matrix &mat)
{
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int num = mat.getValues(i, j);
            setValues(i, j, num);
        }
    }
    m = mat.m;
    n = mat.n;
}
// Matrixiig negeer nemegduuleh;

```

```

void operator++(int)
{
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int sum = getValues(i, j) + 1;
            setValues(i, j, sum);
        }
    }
}

// Matrixiig negeer horogduulah;
void operator--(int)
{
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int sub = getValues(i, j) - 1;
            setValues(i, j, sub);
        }
    }
}

// Matrix deer buhel too nemeed utga olgoh;
void operator+=(int num)
{
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int sum = getValues(i, j) + num;
            setValues(i, j, sum);
        }
    }
}

```

```

// Matrix deer buhel too hasaad utga olgoh;
void operator--=(int num)
{
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int sub = getValues(i, j) - num;
            setValues(i, j, sub);
        }
    }
}

// Matrix deer buhel too urjeed utga olgoh;
void operator*=(int num)
{
    for (int i = 0; i < m; i++)
    {
        for (int j = 0; j < n; j++)
        {
            int res = getValues(i, j) * num;
            setValues(i, j, res);
        }
    }
}

// Tuhain nuden deh matrixiin elementiiг butsaah;
float operator[](int i)
{
    if (m * n ≥ i)
    {
        return values[i];
    }
    return getValues(1, 1);
}

// Matrix hurvuuleh: Matrixiin muriig bagana, baganiig mur болгох;
Matrix operator!()

```

```
{  
    Matrix res(n, m);  
  
    for (int i = 0; i < m; i++)  
    {  
        for (int j = 0; j < n; j++)  
        {  
            int num = getValues(i, j);  
            res.setValues(j, i, num);  
        }  
    }  
    return res;  
}  
};
```