



Цэрэнбат Номин-Эрдэнэ
(20B1NUM0426)

ICSI201 - Объект хандлагат програмчлал

МУИС-ХШУИС
Мэдээллийн технологи

Лаборатори №6

Цэрэнбат Номин-Эрдэнэ

1. ОРШИЛ

Энэ лабораторийн ажил нь матриц классыг тодорхойлон, операторуудыг дахин тодорхойлох

2. ЗОРИЛГО

- Лабораторийн ажлыг алдаагүй зөв хийж гүйцэтгэх.
- Хоёр хэмжээст матрицийг илэрхийлэх класс зарлаж гишүүн өгөгдлүүдийг тохируулах (set)
- утгыг нь буцаах (get)
- байгуулагч функцүүдийг тодорхойлох.

3. ХЭРЭГЖҮҮЛЭЛТ

Хоёр хэмжээст матрицийг илэрхийлэх класс зарлаж гишүүн өгөгдлүүдийг тохируулах (set), утгыг нь буцаах (get) болон байгуулагч функцүүдийг тодорхойл.

```
1  #include<iostream>
2  #include <string.h>
3  #include <math.h>
4
5  using namespace std;
6
7  class matrix {
8  private:
9      int m, n;
10     float values[20][20];
11
12 public:
13     matrix();
14     matrix(int r, int c);
15     ~matrix();
16
17     void print(){
18         for( int i=0; i < m; i++){
19             for ( int j=0 ; j < n; j++){
20                 cout << values[i][j]<<" ";
21             }
22             cout << "\n";
23         }
24     }
25
26     void setRow(int r);
27     void setColumn(int c);
28     int getRow();
29     int getColumn();
30     void nuutsluh();
31     void utgaOnooh();
32     matrix operator +(float f);
33     matrix operator *(matrix a);
34     matrix operator -(matrix a);
35     matrix operator =(matrix a);
36
37     void setRow(int r){
38         delete values;
39         matrix::m=r;
40         nuutsluh();
41     }
42
43     void setColumn(int c){
44         delete values;
45         matrix::n=c;
46         nuutsluh();
47     }
48
49     int getRow(){ //rowiin utga butsaah
50         return m;
51     }
52
53     int getColumn(){ //columniin utga butsaah
54         return n;
55     }
56
57     void nuutsluh(){
58         values = new float*[m];
59         for(int i=0; i<m; i++){
60             values[i] = new float[n];
61             for(int j=0; j<n; j++){
62                 values[i][j] = 0;
63             }
64         }
65     }
66
67     void utgaOnooh(){
68         for(int i=0; i<m; i++){
69             for(int j=0; j<n; j++){
70                 values[i][j] = (float)rand()/(float)RAND_MAX;
71             }
72         }
73     }
74 }
```

Матрицад санах ой нөөцлөх, утга оноох:

```
65
66 matrix::matrix(int r, int c){
67     matrix::m=r;
68     matrix::n=c;
69     nuutsluh();
70     utgaOnooh();
71
72 }
73
74 void matrix::nuutsluh(){
75     int i;
76     values = new float *[m];
77     for(i=0; i<m; i++){
78         values[i]=new float[n];
79     }
80 }
81
82 void matrix::utgaOnooh(){
83     int i, j;
84     for(i=0; i<m; i++){
85         for(j=0; j<n; j++){
86             values[i][j]=rand(n)%10;
87         }
88     }
89 }
```

Устгагч функц:

```
90
91 matrix::~~matrix(){
92     delete values;
93 }
94
```

1. + нэмэх(хоёр матрицыг нэмнэ), хоёр янзаар тодорхойлно: а) $R = R1 + 12.5(\text{float})$ б) $R = R1 + R2$

```
97
98 matrix matrix::operator +(float f){
99     int i, j;
100     matrix t(m,n);
101     for(i=0; i<m; i++){
102         for(j=0; j<n; j++){
103             t.values[i][j] = values[i][j] + f;
104         }
105     }
106     return t;
107 }
108
109 int main(){
110     srand(time(NULL));
111     matrix m1(3, 4);
112     m1.print();
113     cout<<"+"<<endl;
114     cout<<"14"<<endl;
115     cout<<"="<<endl;
116     m1=m1+14;
117     m1.print();
118     return 0;
119 }
```

```
87
88 //-----1-----
89 //nemeh b) R = R1 + R2
90
91 matrix matrix::operator +(matrix a){
92     int i, j;
93     matrix t(m,n);
94     for(i=0; i<m; i++){
95         for(j=0; j<n; j++){
96             t.values[i][j] = values[i][j] + a.values[i][j];
97         }
98     }
99     return t;
100 }
101
102 int main(){
103     srand(time(NULL));
104     matrix m1(3, 4), m2(3, 4);
105     m1.print();
106     cout<<"+"<<endl;
107     m1.print();
108     cout<<"="<<endl;
109     m1=m1+m2;
110     m1.print();
111     return 0;
112 }
```

2. * үржих (хоёр матрицийг үржээд үржвэрийг буцаана)

```
97
98 matrix matrix::operator *(matrix a){
99     int i, j, k;
100     matrix t(m, a.n);
101     if(n==a.m){
102         for(i=0; i<m; i++){
103             for(j=0; j<a.n; j++){
104                 for(k=0; k<n; k++){
105                     t.values[i][j] += values[i][k] * a.values[k][j];
106                 }
107             }
108         }
109         return t;
110     }
111     else{
112         cout<<"urjuuleh bolomjgui"<<endl;
113         return -1;
114     }
115 }
116
117 int main(){
118     srand(time(NULL));
119     matrix m1(3, 4), m2(4, 4), m3(3, 4);
120     m1.print();
121     cout<<"*"<<endl;
122     m2.print();
123     cout<<"="<<endl;
124     m3=m1*m2;
125     m3.print();
126     return 0;
127 }
```

3. - хасах (хоёр матрицийг хасаад ялгаварыг буцаана)

```
98 matrix matrix::operator -(matrix a){
99     int i, j;
100     matrix t(m, n);
101     for(i=0; i<m; i++){
102         for(j=0; j<n; j++){
103             t.values[i][j] = values[i][j] - a.values[i][j];
104         }
105     }
106     return t;
107 }
108
109
110 int main(){
111     srand(time(NULL));
112     matrix m1(3, 4), m2(4, 4);
113     m1.print();
114     cout<<"-"<<endl;
115     m2.print();
116     cout<<"="<<endl;
117     m1=m1-m2;
118     m1.print();
119     return 0;
120 }
```

6. = нэг матрицийн объектыг нөгөөд утга оноох

```
97
98 = matrix matrix::operator =(matrix a){
99     int i, j;
100     matrix t(m, n);
101 =     for(i=0; i<a.getRow; i++){
102 =         for(j=0; j<a.getColumn; j++){
103             values[i][j] = a.values[i][j];
104         }
105     }
106     return t;
107 }
108
109
110 = int main(){
111     srand(time(NULL));
112     matrix m1(3, 4), m2(4, 4);
113     m1.print();
114     m2.print();
115     cout<<"="<<endl;
116     m1=m2;
117     m1.print();
118     return 0;
119 }
```

7. ++ нэгээр нэмэгдүүлэх

```
97
98 = matrix matrix::operator ++(int){
99     int i, j;
100     matrix t(m, n);
101 =     for(i=0; i<m; i++){
102 =         for(j=0; j<n; j++){
103             values[i][j]++;
104         }
105     }
106     return t;
107 }
108
109
110 = int main(){
111     srand(time(NULL));
112     matrix m1(3, 4);
113     m1.print();
114     m1++;
115     cout<<endl;
116     m1.print();
117     return 0;
118 }
```

8. -- нэгээр хорогдуулах

```
97
98 matrix matrix::operator --(int){
99     int i, j;
100     matrix t(m, n);
101     for(i=0; i<m; i++){
102         for(j=0; j<n; j++){
103             values[i][j]--;
104         }
105     }
106     return t;
107 }
108
109
110 int main(){
111     srand(time(NULL));
112     matrix m1(3, 4);
113     m1.print();
114     m1--;
115     cout<<endl;
116     m1.print();
117     return 0;
118 }
```

9. +=

```
97
98 matrix matrix::operator +=(int a){
99     int i, j;
100     matrix t(m, n);
101     for(i=0; i<m; i++){
102         for(j=0; j<n; j++){
103             values[i][j]+=a;
104         }
105     }
106 }
107
108
109 int main(){
110     srand(time(NULL));
111     matrix m1(3, 4);
112     m1.print();
113     m1+=4;
114     cout<<endl;
115     m1.print();
116     return 0;
117 }
```


10. -=

```
97
98 matrix matrix::operator -=(int a){
99     int i, j;
100     matrix t(m, n);
101     for(i=0; i<m; i++){
102         for(j=0; j<n; j++){
103             values[i][j]-=a;
104         }
105     }
106 }
107
108
109 int main(){
110     srand(time(NULL));
111     matrix m1(3, 4);
112     m1.print();
113     m1-=4;
114     cout<<endl;
115     m1.print();
116     return 0;
117 }
```

11. *=

```
97
98 matrix matrix::operator *=(int a){
99     int i, j;
100     matrix t(m, n);
101     for(i=0; i<m; i++){
102         for(j=0; j<n; j++){
103             values[i][j]*=a;
104         }
105     }
106 }
107
108
109 int main(){
110     srand(time(NULL));
111     matrix m1(3, 4);
112     m1.print();
113     m1*=4;
114     cout<<endl;
115     m1.print();
116     return 0;
117 }
```