

```
1 #ifdef _WIN32
2     #pragma warning(disable:4996)
3     // #include <tchar.h>
4     // #include <windows.h>
5     #include <conio.h>
6     #include <direct.h>
7 #elif (defined __linux__) || (defined _AIX)
8     #include <stdlib.h>
9     #include <sys/types.h>
10    #include <sys/stat.h>
11    #include <unistd.h>
12    typedef char _TCHAR;
13    #define _tmain main
14 #endif
15
16 #include <stdio.h>
17 #include <iostream>
18 #include <iomanip>
19 using namespace std;
20
21 #include "MaticaObdlznikova.h"
22 #include "MaticaStvorcova.h"
23 #include "CisloKomplexne.h"
24 #include "MaticaStvorcovaKomplexna.h"
25 //-----
26 TMaticaStvorcovaKomplexna::TMaticaStvorcovaKomplexna()
27 {
28     TCisloKomplexne zero(0,0);
29
30     PocetRiadkov=MAXPOCET;
31     PocetStlpcov=MAXPOCET;
32     for (unsigned i=0; i<PocetRiadkov; i++)
33         for (unsigned j=0; j<PocetStlpcov; j++)
34             Matica[i][j]=zero;
35 }
36 //-----
37 TMaticaStvorcovaKomplexna::TMaticaStvorcovaKomplexna(TMaticaStvorcovaKomplexna& X)
38 {
39     PocetRiadkov = X.PocetRiadkov;
40     PocetStlpcov = X.PocetStlpcov;
41     for (unsigned i=0; i<PocetRiadkov; i++)
42         for (unsigned j=0; j<PocetStlpcov; j++)
43             Matica[i][j] = X.Matica[i][j];
44 }
45 //-----
46 TMaticaStvorcovaKomplexna::~TMaticaStvorcovaKomplexna()
47 {
48 }
49 //-----
50 TMaticaStvorcovaKomplexna& TMaticaStvorcovaKomplexna::operator=
51     (const TMaticaStvorcovaKomplexna& X)
52 {
53     if (this == &X) return *this;
```

```

54   PocetRiadkov = X.PocetRiadkov;
55   PocetStlpcov = X.PocetStlpcov;
56   for (unsigned i=0;i<PocetRiadkov;i++)
57       for (unsigned j=0;j<PocetStlpcov;j++)
58           Matica[i][j] = X.Matica[i][j];
59   return *this;
60 }
61 //-----
62 const TMaticaStvorcovaKomplexna operator+(const TMaticaStvorcovaKomplexna&
63   LavaMatica,const TMaticaStvorcovaKomplexna& PravaMatica)
64 {
65     TMaticaStvorcovaKomplexna VyslMatica;
66
67     VyslMatica.PocetRiadkov = LavaMatica.PocetRiadkov;
68     VyslMatica.PocetStlpcov = LavaMatica.PocetStlpcov;
69     for (unsigned i=0; i<LavaMatica.PocetRiadkov; i++)
70         for (unsigned j=0; j<LavaMatica.PocetStlpcov; j++)
71             VyslMatica.Matica[i][j] = LavaMatica.Matica[i][j]+PravaMatica.Matica[i][j];
72     return VyslMatica;
73 }
74 //-----
75 const TMaticaStvorcovaKomplexna operator*(const TMaticaStvorcovaKomplexna&
76   LavaMatica,const TMaticaStvorcovaKomplexna& PravaMatica)
77 {
78     TMaticaStvorcovaKomplexna VyslMatica;
79     my_class xx;
80     unsigned i,j,k;
81     TCisloKomplexne Suma(0,0);
82
83     if (LavaMatica.PocetStlpcov!=PravaMatica.PocetRiadkov){
84         cout << "\nMatice sa nedaju nasobit!";
85         xx.my_getch();
86         exit(1);
87     }
88
89     VyslMatica.PocetRiadkov=LavaMatica.PocetRiadkov;
90     VyslMatica.PocetStlpcov=PravaMatica.PocetStlpcov;
91     for (i=0; i<LavaMatica.PocetRiadkov; i++)
92         for (j=0; j<PravaMatica.PocetStlpcov; j++){
93             for (k=0; k<LavaMatica.PocetStlpcov; k++) {
94                 Suma = Suma + LavaMatica.Matica[i][k] * PravaMatica.Matica[k][j];
95                 /* Suma.Re = Suma.Re + LavaMatica.Matica[i][k].Re * PravaMatica.Matica[k][j].Re
96                    - LavaMatica.Matica[i][k].Im * PravaMatica.Matica[k][j].Im;
97                 Suma.Im = Suma.Im + LavaMatica.Matica[i][k].Re * PravaMatica.Matica[k][j].Im
98                    + LavaMatica.Matica[i][k].Im * PravaMatica.Matica[k][j].Re; */
99             }
100            VyslMatica.Matica[i][j]=Suma;
101        }
102
103     return VyslMatica;
104 }
105 //-----
106 istream& operator>>(istream& is, TMaticaStvorcovaKomplexna& X)

```

```
107 {
108     my_class xx;
109
110     is >> X.PocetRiadkov >> X.PocetStlpcov;;
111     if (is.fail()){
112         cout << "Subor MATICA.TXT sa nepodarilo otvorit.";
113         xx.my_getch();
114         exit(1);
115     }
116     for (unsigned i=0;i<X.PocetRiadkov;i++)
117         for (unsigned j=0;j<X.PocetStlpcov;j++)
118             is >> X.Matica[i][j];
119
120     return is;
121 }
122 //-----
123 ostream& operator<<(ostream& os,const TMaticaStvorcovaKomplexna& X)
124 {
125     os.setf(ios::fixed,ios::floatfield);
126     os.precision(2);
127     for (unsigned i=0;i<X.PocetRiadkov;i++) {
128         os << "\n ";
129         for (unsigned j=0;j<X.PocetStlpcov;j++)
130             os << X.Matica[i][j] << " ";
131     }
132     os << "\n";
133
134     return os;
135 }
136 //-----
137 void not_child_class::not_child_method()
138 {
139     // TMaticaStvorcovaKomplexna A,zero(0, 0);
140     // A = zero;
141 }
142 //-----
```