

```
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 //-----
24 TMaticaStvorcova::TMaticaStvorcova()
25 {
26 }
27 //-----
28 TMaticaStvorcova::TMaticaStvorcova(TMaticaStvorcova& X)
29 {
30     PocetRiadkov = X.PocetRiadkov;
31     PocetStlpcov = X.PocetStlpcov;
32     for (unsigned i = 0; i<PocetRiadkov; i++)
33         for (unsigned j = 0; j<PocetStlpcov; j++)
34             Matica[i][j] = X.Matica[i][j];
35 }
36 //-----
37 TMaticaStvorcova::~TMaticaStvorcova()
38 {
39 }
40 //-----
41 TMaticaStvorcova& TMaticaStvorcova::operator=(const TMaticaStvorcova& X)
42 {
43     if (this == &X) return *this;
44
45     PocetRiadkov = X.PocetRiadkov;
46     PocetStlpcov = X.PocetStlpcov;
47     for (unsigned i = 0; i<PocetRiadkov; i++)
48         for (unsigned j = 0; j<PocetStlpcov; j++)
49             Matica[i][j] = X.Matica[i][j];
50
51     return *this;
52 }
53 //-----
```

```
54 TMaticaStvorcova& TMaticaStvorcova::operator=(const TMaticaObdlznikova& X)
55 {
56     if (this == &X) return *this;
57
58     my_class xx;
59
60     if (X.getPocetRiadkov() != X.getPocetStlpcov()) {
61         cout << "\\nTato obdlznikova matica sa neda previest na stvorcovu!";
62         xx.my_getch();
63         exit(1);
64     }
65
66     PocetRiadkov = X.getPocetRiadkov();
67     PocetStlpcov = X.getPocetStlpcov();
68     for (unsigned i = 0; i<PocetRiadkov; i++)
69         for (unsigned j = 0; j<PocetStlpcov; j++)
70             Matica[i][j] = X.getPrvokMatice(i, j);
71
72     return *this;
73 }
74 //-----
75 TMaticaStvorcova TMaticaStvorcova::UmocniMaticu(unsigned k)
76 {
77     TMaticaStvorcova Y, PomMatica0, PomMatica;
78     my_class x;
79
80     if (PocetStlpcov != PocetRiadkov) {
81         cout << "\\n Matica sa neda umocnit!";
82         x.my_getch();
83         exit(1);
84     }
85     Y.PocetRiadkov = Y.PocetStlpcov = PocetRiadkov;
86
87     switch (k) {
88     case 0:
89         for (unsigned i = 0; i<PocetRiadkov; i++) {
90             for (unsigned j = 0; j<PocetRiadkov; j++)
91                 Y.Matica[i][j] = 0;
92             Y.Matica[i][i] = 1;
93         }
94         break;
95     case 1:
96         Y = *this;
97         break;
98     default:
99         PomMatica0 = PomMatica = *this;
100        k--;
101        while (k-->0) {
102            Y = PomMatica * PomMatica0;
103            PomMatica = Y;
104        }
105    }
106}
```

107 return Y;

108 }

109 //-----

110