

```

1 trida.cpp
2 //Vlozeni hlavickoveho souboru, pokud jeste neni vlozeny
3 #include "trida.h"
4 //Konstuktor tridy Trida = pocatecni nastaveni "hodnot" promennych tridy Trida,
5 //pokud tridu nepretezim
6 Trida::Trida(){
7     this->PrvniPrivatniPromenna = 0.0;
8     this->DruhaPrivatniPromenna = 0.0;
9 }
10 //Pretizeni tridy Trida
11 Trida::Trida(float prvniPrivatniPromenna){
12     this->PrvniPrivatniPromenna = prvniPrivatniPromenna;
13     this->DruhaPrivatniPromenna = 0.0;
14 }
15
16 //Pretizeni tridy Trida
17 Trida::Trida(float prvniPromennaParametru, float druhaPromennaParametru){
18     this->PrvniPrivatniPromenna = prvniPromennaParametru;
19     this->DruhaPrivatniPromenna = druhaPromennaParametru;
20 }
21
22 Trida Trida::operator+(Trida cisloProPridani){
23     //Vytvoreni jakoby promenne typu Trida
24     Trida vysledek;
25     vysledek.PrvniPrivatniPromenna = this->PrvniPrivatniPromenna +
26     cisloProPridani.PrvniPrivatniPromenna;
27     vysledek.DruhaPrivatniPromenna = this->DruhaPrivatniPromenna +
28     cisloProPridani.DruhaPrivatniPromenna;
29     return vysledek;
30 }
31 //Pretizeni operatoru << (pro vystup)
32 ostream &operator<<(ostream &naVystup, Trida &x){
33     cout<<"Prvni promenna | Druha promenna: ";
34     naVystup<<x.PrvniPrivatniPromenna<<"|"<<x.DruhaPrivatniPromenna<<endl;
35 }
36 //Pretizeni operatoru >> (pro vstup)
37 istream &operator>>(istream &naVstup, Trida &x){
38     cout<<"Zadejte prvni privatni promennou: ";
39     naVstup>>x.PrvniPrivatniPromenna;
40     cout<<"Zadejte druhou privatni promennou: ";
41     naVstup>>x.DruhaPrivatniPromenna;
42 }
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71

```

```

72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88 main.cpp
89 //Vložení hlavičkového souboru, pokud ještě není vložený
90 #include "trida.h"
91
92 int main() {
93     //Vytvoření více instancí třídy Trida
94     //Podobně jako vytvoření více proměnných typu ...,
95     //jedna se ale o objekt!
96     Trida prvni, druha, treti;
97     //Různé typy vytvoření, pozitivní přetečení třídy:
98     Trida ctvrta(0.5);
99     Trida pata(0.99,1.20);
100
101     druha = ctvrta+pata;
102
103     //Použití operátoru << pro výpis
104     cout<<prvni;
105     cout<<treti;
106     cout<<druha;
107     cout<<pata;
108     return 0;
109 }
110
111 trida.h
112 #ifndef _TRIDA //Vloží se jen pokud již není vložena
113 #define _TRIDA
114
115 #include <iostream>
116 #include <iomanip>
117 #include <cstdlib>
118
119 //Jmenný prostor, abychom nemuseli psát std::
120 using namespace std;
121
122 class Trida{
123     private:
124         float PrvniPrivatniPromenna;
125         float DruhaPrivatniPromenna;
126     public:
127         //Konstruktor (odkaz na trida.cpp)
128         Trida();
129         //První přetížení
130         Trida(float prvniPromennaParametru);
131         //Druhé přetížení
132         Trida(float prvniPromennaParametru, float druhaPromennaParametru);
133         //Vlastní veřejná metoda - nic nenastavuje => const
134         void PrvniVeřejnaMetoda() const;
135         //Přetížení operátoru +
136         Trida operator+(Trida cisloProPřidání);
137         //Deklarace přetížení operátoru <<
138         friend ostream &operator<<(ostream &naVystup, Trida &x);
139         //Deklarace přetížení operátoru >>
140         friend istream &operator>>(istream &naVstup, Trida &x);
141 };
142 #endif
143
144

```