

```

1  #ifdef _WIN32
2      #pragma warning(disable:4996)
3      #include <tchar.h>
4      #include <conio.h>
5  #elif (defined __linux__ ) || (defined _AIX) || (defined __APPLE__ )
6      typedef char _TCHAR;
7      #define _tmain main
8  #endif
9
10 #include <stdio.h>
11 #include <string.h>
12 #include <iostream>
13 using namespace std;
14
15 #include "Veta.h"
16 //-----
17 VetaClass::VetaClass ()
18 {
19     for (unsigned i = 0; i < 150; i++) {
20         strcpy(Slova[i].InterpunkciaPredSlovom, "");
21         Slova[i].Slovo = new char[40];
22         strcpy(Slova[i].Slovo, "");
23         strcpy(Slova[i].InterpunkciaPoSlove, "");
24     }
25 }
26
27 VetaClass::~VetaClass ()
28 {
29     for(unsigned i=0;i<150;i++)
30         delete[] Slova[i].Slovo;
31 }
32
33 istream& operator>>(istream& is, VetaClass* x)
34 {
35     unsigned nove_slovo, cislo_slova;
36     char c;
37
38     //Vyhadzujeme medery medzi slovami a triedime vetu do slov.
39     nove_slovo = 0;
40     cislo_slova = 0;
41     do {
42         is.get(c);
43         if (isspace(c)) {
44             cislo_slova++;
45             nove_slovo = 1;
46             do {
47                 is.get(c);
48             } while ((!isalnum(c)) && (!ispunct(c)));
49         }
50         if ((ispunct(c)) && (c != '-')) {
51             if (nove_slovo)
52                 strncat(x->Slova[cislo_slova].InterpunkciaPredSlovom, &c, 1);
53             else
54                 strncat(x->Slova[cislo_slova].InterpunkciaPoSlove, &c, 1);
55             nove_slovo = 0;
56         }
57         if ((isalnum(c)) || (c == '-')) {
58             strncat(x->Slova[cislo_slova].Slovo, &c, 1);
59             nove_slovo = 0;
60         }
61     } while ((c != '.') && (c != '!') && (c != '?'));
62     x->PocetSlov = ++cislo_slova;
63     return is;
64 }
65
66 ostream& operator<<(ostream& os, VetaClass* x)
67 {
68     for (unsigned i = 0; i < x->PocetSlov; i++) {
69         os << x->Slova[i].InterpunkciaPredSlovom;
70         os << x->Slova[i].Slovo;
71         os << x->Slova[i].InterpunkciaPoSlove;
72         os << " ";
73     }

```

```
74     return os;  
75 }  
76  
77 char* VetaClass::DajSlovo(unsigned cislo_slova)  
78 {  
79     return this->Slova[cislo_slova].Slovo;  
80 }
```