

## Kolokwium Testowe 1, gr. A, Języki Programowania, 13.11.2011

### Treść zadania

Należy stworzyć klasę Owoc zawierającą trzy pola składowe:

- nazwa(char \*) //ilość znaków dla pola nazwa powinna być dowolna
- ilosc (int)
- cena (double)

Oraz zestaw metod: metody zwracające i ustawiające składniki klasy (metody typu "get" i "set"), konstruktor domyślny, konstruktor z 3 parametrami oraz konstruktor kopiujący i destruktor. Należy również przeciążyć operator przekierowania ("<<"). Ponadto należy stworzyć metodę SetAll ustawiającą wszystkie trzy parametry jednocześnie (jeśli podamy dwa parametry, wtedy cena=1.0, a jeśli tylko jeden parametr, wtedy cena=1.0, a ilosc = 0). Program powinien się kompilować bez żadnych ostrzeżeń.

Klasy powinny być skonstruowane w taki sposób, by działały z podanym niżej programem:

```
#include <iostream>
#include <cstring>
#include "janik.h"

using namespace std;

int main()
{
    char* gruszka = new char[8];    strcpy(gruszka, "gruszka");
    char* jablko= new char[7];     strcpy(jablko, "jablko");
    char* banan= new char[6];      strcpy(banan, "banan");

    Owoc o1;
    o1.SetNazwa(gruszka);
    cout<<"Nazwa: "<<o1.GetNazwa()<<" Ilocz: "<<o1.GetIlocz()<<" Cena: "<<o1.GetCena()<<endl;

    Owoc *o2 = new Owoc(jablko,10,1.1);
    cout<<"Nazwa: "<<o2->GetNazwa()<<" Ilocz: "<<o2->GetIlocz()<<" Cena: "<<o2->GetCena()<<endl;
    cout<<o1<<" ", "<<*o2<<endl;
    delete o2;
    o1.SetAll(banan);
    cout<<o1<<endl;
    o1.SetAll(banan,2);
    cout<<o1<<endl;
    o1.SetAll(banan,2,4.5);
    cout<<o1<<endl;

    return 0;
}
```

Co po uruchomieniu skutkowało by pojawieniem się na ekranie:

```
$ ./program
Nazwa: gruszka Ilocz: 0 Cena: 1
Nazwa: jablko Ilocz: 10 Cena: 1.1
Owoc: (gruszka, 0, 1), Owoc: (jablko, 10, 1.1)
Jestem w destruktorze..
Owoc: (banan, 0, 1)
Owoc: (banan, 2, 1)
Owoc: (banan, 2, 4.5)
Jestem w destruktorze..
```

Program, oraz wszystkie pliki należy wysłać przed końcem zajęć na adres [majanik@if.pw.edu.pl](mailto:majanik@if.pw.edu.pl), a nazwy plików z klasą owoc powinny zawierać nazwisko np: kowalski.cpp oraz kowalski.h.

## Kolokwium Testowe 1, gr. B, Języki Programowania, 13.11.2011

### Treść zadania

Należy stworzyć klasę vector zawierającą pola składowe:

- x, y, z (double)
- notatka (std::string)
- pole statyczne ilosc (int)

Oraz zestaw metod: metodę zwracającą i ustawiającą notatkę (metodę typu "get" i "set"), konstruktor domyślny, konstruktor z 3 parametrami (x,y,z), konstruktor kopiujący i destruktor. Konstruktory prócz przypisywania wartości odpowiednim składnikom powinny zwiększać wartość pola ilość o 1, destruktor powinien zmniejszać wartość tego pola. Należy również stworzyć metodę statyczną zwracającą wartość pola ilosc, oraz przeciążyć operator + (zwracający wektor) oraz [ ] (by zwracał x, y oraz z dla odpowiednio wartości 0, 1, > 1). Oraz metodę Dlugosc() zwracającą długość wektora. Ponadto należy stworzyć metodę SetXYZ ustawiającą trzy parametry (x, y, z) jednocześnie oraz stworzyć metodę WypiszWektor() która wypisuje na ekran informację o składnikach danego obiektu. Do tego napisać metodę WypiszWektory przyjmującą tablicę wektorów oraz ich liczbę, służącą do wypisywania owej podanej tablicy na ekran.

Program powinien się kompilować bez żadnych ostrzeżeń.

Klasy powinny być skonstruowane w taki sposób, by działały z podanym niżej programem :

```
#include <iostream>
#include "vector.h"
using namespace std;

int main()
{
    vector v;
    v.WypiszWektor(); cout<<endl;
    vector *v2 = new vector(2,3,4);
    v2->WypiszWektor(); cout<<endl;
    v.SetXYZ(1,2,3);
    v.SetNotatka("wektor pierwszy");
    v2->SetNotatka("wektor drugi");
    v.WypiszWektor(); cout<<endl;
    v2->WypiszWektor(); cout<<endl;
    vector v3 = v;
    v3 = v3 + *v2;
    v3.WypiszWektor(); cout<<"Dlugosc wektora v3: "<<v3.Dlugosc()<<endl;
    cout<<"Liczba wektorow: "<<vector::GetIlosc()<<endl;
    delete v2;
    cout<<"Liczba wektorow: "<<vector::GetIlosc()<<endl;
    v3[0]=6; cout<<v3[0]<<" "<<v3[1]<<endl;
    vector tabV[2]; tabV[0].SetXYZ(6,6,6); tabV[1].SetXYZ(2,2,2);
    WypiszWektory(tabV,2);
    return 0;
}
```

Co po uruchomieniu skutkowało by pojawieniem się na ekranie:

```
$ ./program
(0,0,0)
(2,3,4)
(1,2,3) wektor pierwszy
(2,3,4) wektor drugi
(3,5,7) Dlugosc wektora v3: 9.11043
Liczba wektorow: 3
Liczba wektorow: 2
6 5
(6,6,6)
(2,2,2)
```

Program należy wysłać przed końcem zajęć na adres [majanik@if.pw.edu.pl](mailto:majanik@if.pw.edu.pl), a nazwy plików z klasą wektor powinny zawierać nazwisko np: kowalski.cpp oraz kowalski.h.