

Kolokwium Testowe 1, gr. A, Języki Programowania

Treść zadania

Należy stworzyć klasę Owoc zawierającą trzy pola składowe:

- nazwa(char *) //zakładamy, że nazwa nie przekroczy 25 znaków, ale pamięć powinna być alokowana dynamicznie
- ilosc (liczba całkowita)
- cena (liczba zmiennoprzecinkowa)

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 zewnętrzną funkcję Wypisz() która przyjmuje obiekt Owoc i wypisuje go na ekran . 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), oraz przeciążyć operator przekierowania: (“<<”).

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

(program można ściągnąć ze strony: http://www.if.pw.edu.pl/~majanik/data/JP/2013/kolokwium_old/mainowoc.cpp)

```
#include <iostream>
#include <cstring>
#include "owoc.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;
    delete gruszka;
    Wypisz(o1);

    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;

    Owoc o3(o1); Wypisz(o3);
    o3.SetAll(banan);
    cout<<o3<<endl;
    o3.SetAll(banan,2);
    cout<<o3<<endl;
    o3.SetAll(banan,2,4.5);
    return 0;
}
```

Co po uruchomieniu skutkowało by pojawieniem się na ekranie:

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

Plik nagłówkowy powinien zostać otoczony strukturą „ifndef”. Program powinien być napisany w 3 plikach (**owoc.cpp**, **owoc.h** oraz **program.cpp** z funkcją main()) oraz powinien się kompilować bez żadnych ostrzeżeń (flaga -Wall).

Program, oraz wszystkie pliki należy wysłać przed końcem zajęć na adres majanik@if.pw.edu.pl.

Kolokwium Testowe 1, gr. D, Języki Programowania

Należy stworzyć klasę Czlowiek, zawierającą następujące pola składowe:

Czlowiek:

- std::string imie
- std::string nazwisko
- int* pesel //przechowującą numer PESEL (11 liczb, że każdy int jest cyfrą 0-9)

Oraz zestaw metod:

konstruktor domyślny (domyślny właściciel to "John Smith, 000000000"), konstruktor z 3 parametrami, konstruktor kopiujący i destruktor, oraz:

- metody ustawiające i zwracające składniki klasy (metody typu "set" i „get”), metoda „set” dla operator<<, pozwalający na wypisanie wszystkich pól składowych na ekran
- operator[], umożliwiający zmianę/odzyskanie zadanej cyfry numeru pesel
- operator=, zgodnie ze zdrowym rozsądkiem
- do tego napisać funkcję (zewnętrzna) WypiszLudzi przyjmującą tablicę obiektów Czlowiek oraz ich liczbę, służącą do wypisywania owej podanej tablicy na ekran

Klasy powinny być skonstruowane w taki sposób, by działały z podanym niżej programem (do ściągnięcia ze strony:

http://www.if.pw.edu.pl/~majanik/data/JP/2012/Kolokwium/program_wt_2.cpp):

```
#include <iostream>
#include "Czlowiek.h"

using namespace std;

int main()
{
    Czlowiek cz0;
    cout<<cz0<<endl;

    int pesel[]={0,0,0,0,0,0,1,1,0,0,7};
    Czlowiek cz1("James","Bond",pesel);
    cout<<cz1<<endl;

    Czlowiek *cz2 = new Czlowiek(cz1);
    cz2->SetNazwisko("Black");
    cout<<*cz2<<endl;

    int pesel2[]={2,2,2,2,2,2,5,5,3,3,3};
    cz1.SetPesel(pesel2);
    cout<<cz1[2]<<" "<<cz1[7]<<" "<<cz1[10];
    cout<<endl;

    Czlowiek tab[]={cz0, cz1, *cz2};
    WypiszLudzi(tab,3);

    return 0;
}
```

Co po uruchomieniu skutkowało by pojawieniem się na ekranie:

```
$ ./program
Czlowiek: John Smith: 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Czlowiek: James Bond: 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 7
Czlowiek: James Black: 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 7
2 5 3
Czlowiek: John Smith: 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Czlowiek: James Bond: 2, 2, 2, 2, 2, 2, 5, 5, 3, 3, 3
Czlowiek: James Black: 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 7
```

Plik nagłówkowy powinien zostać otoczony strukturą „ifndef”. Program powinien być napisany w 2 plikach (czlowiek.h, czlowiek.cpp oraz program.cpp z funkcją main()) oraz powinien się kompilować bez żadnych ostrzeżeń (flaga -Wall).

Program, oraz wszystkie pliki należy wysłać przed końcem zajęć na adres majanik@if.pw.edu.pl! Uwaga, za niekompilujący się program można uzyskać co najwyżej 40% punktów!