

Zaawansowane programowanie C++

Lista 9 - biblioteki

Zadanie 17

Singleton, to klasa, której można utworzyć tylko jedną instancję.

Napisz niekopiowalną klasę `Singleton`, która będzie składać się z:

- prywatnego konstruktora - wypisującego adres tworzonego obiektu i znak '+',
- destruktora - wypisującego adres niszczonego obiektu i znak '-',
- metody statycznej `instance()` - zwracającej referencję do jedynej instancji klasy, utworzonej przy pierwszym wywołaniu `instance()`,
- Metody `calls()`, wypisującej adres obiektu i ilość wywołań na nim metody `calls()`.

Deklarację klasy `Singleton` umieść w pliku `singleton.h`,
a definicje wszystkich jej metod w pliku `singleton.cpp`.

Dodatkowo w pliku `calls1.h` napisz deklarację funkcji `calls1()`, a w pliku `calls1.cpp` jej definicję, która na jedynej instancji klasy `Singleton` wywoła metodę `calls()`.

Z powyższych plików utwórz bibliotekę statyczną o nazwie `singleton_static`.

Napisz program, który przetestuje działanie klasy `Singleton` tworząc jej instancję iwołając na niej metodę `calls()`, a następniewołając funkcję `calls1()`.
Program powinien składać się z jednego pliku źródłowego `main.cpp` i używać biblioteki `singleton_static`.

Stwórz bibliotekę dynamiczną `singleton_dynamic`, która będzie zawierać dwie funkcje:

- `calls2()` - robiącą to samo co `calls1()` z biblioteki `singleton_static`,
- `instance()` - zwracającą referencję do obiektu klasy `Singleton`.

Biblioteka `singleton_dynamic` powinna składać się z plików: `singleton_dynamic.h` i `singleton_dynamic.cpp` oraz `singleton.h` i `singleton.cpp`.

Zlinkuj program testowy z biblioteką `singleton_dynamic` i rozszerz go o wywołania funkcji `calls2()` oraz metody `calls()` na obiekcie zwróconym przez funkcję `instance()`.