

Utworzyć klasę A, zawierającą wiersz tekstowy char * str.

Utworzyć klasę B, która zawiera zmienną int i.

Utworzyć klasę C, która dziedziczy po klasie A i klasie B: class C : public A, public B {...}. Swoich własnych zmiennych klasa C nie ma.

Klasy A, B, C powinny zawierać konstruktory, destruktory (jeśli jest potrzebny) oraz wszystkie niezbędne metody, które są potrzebne dla wykonania kodu funkcji main().

Utworzyć funkcje-szablon

template <class T> void PrintToScreen(const T& ob, const char* title), gdzie ob – obiekt dowolnego typu, title – wiersz tekstowy.

Przystosować klasy A, B, C do poprawnego działania w kodzie:

```
int main()
{
    vector<CC> vect;

    CC tab_C[] = { CC(1, "aaa"), CC(2, "ccc"), CC(3, "bbb"), CC(4, "ccc"), CC(5, "bbb") };

    const size_t no_elem = sizeof(tab_C) / sizeof(CC);

    //Put elements of tab_C in vect

    //output elements of vect to screen
    PrintToScreen(vect, "before sorting");

    //sort elements of vector according titles using algorithm std::sort

    //output elements of vect to screen
    PrintToScreen(vect, "after sorting");
}
```

```
        //remove all elements from vector vect with the duplicated titles using algorithm std::unique oraz metodę wektora erase.  
  
        //output elements of vect to screen  
        PrintToScreen(vect, "finally:");  
  
        vect.clear();  
    }
```

Sprawdzić, czy wywołuje się destruktor klasy A. Po zakończeniu proszę przesłać archiwum projektu przez czat MS TEAMS.