

Zadanie na Lab. 1

Zadanie I (za 6 pkt.):

Zadanie

Zadeklaruj kilkuelementową tablicę, korzystając z szablonu **Tablica** prezentowanego na wykładzie z PO. Kod szablonu jest dostępny na stronie [www](http://www.uksw.edu.pl):

[Kody programów \[PO\] – Krzysztof Trojanowski \(uksw.edu.pl\)](http://www.uksw.edu.pl)

sekcja „Program #13”, plik wykład13b.h

Zadeklaruj klasę konkretyzując szablon Tablica, która przechowuje elementy typu:

```
struct Dane {  
    int ID;  
    char nick[10];  
    unsigned level;  
    double pkt;  
};
```

W funkcji main utwórz obiekt tej klasy i wypełnij kilka pierwszych komórek tablicy rozsądnymi wartościami (podobnie jak to zostało pokazane w sekcji „Program #13”, plik main.cpp).

Następnie napisz funkcję, która jako argumenty przyjmuje referencję do obiektu tablicowego oraz nazwę pliku i tryb dostępu do pliku. Funkcja generuje do wskazanego pliku tekstowego we wskazanym trybie dostępu (tryb „utwórz nowy plik” lub „dołącz do istniejącego”) raport, zawierający wszystkie dane z tablicy. Uwaga: funkcja sprawdza, czy podany tryb jest właściwy dla czynności pisania. Jeżeli nie – natychmiast kończy swoje działanie.

Raport ma następującą budowę:

1. nagłówek, zawierający datę i czas wyrównaną do prawej (aktualna data i czas zostaje ustalona za pomocą funkcji odczytu czasu z biblioteki `<ctime>`) oraz nazwę komputera wyrównaną do lewej (można ją pobrać ze zmiennej systemowej `COMPUTERNAME`), linię poziomą rozdzielającą, linię z nagłówkami dla kolumn tabelki i drugą linię poziomą rozdzielającą (ramki i nazwy kolumn są zapisane w kodzie w programie na sztywno, o treści takiej, jak np. w przykładzie poniżej),
2. treść, zawierająca tabelkę, w której kolejne wiersze reprezentują elementy z tablicy,
3. stopka, zawierająca: linię odcinającą tabelkę, oraz wiersz zawierający: zerową wartość w polu ID i pustą w nick, największą wartość level i wartość pkt dla tego elementu tablicy z największą wartością level (do znalezienia największej wartości level dla typu **Dane**, którego instancje będą przechowywane w tablicy, napisz przeciążony operator porównania).

Przykładowe dane do tablicy może podać użytkownik, lub mogą być zapisane w kodzie funkcji main. Program pyta użytkownika o nazwę pliku do zapisu.

Uwaga 1: Wiersz tabelki powinien być generowany za pomocą przeciążonego operatora pisania do strumienia dla typu **Dane** (zobacz slajdy 344-346, wykład 15 PO) i być od razu sformatowany tak, aby zachowana była szerokość kolumn oraz format postaci tekstowej wartości pól:

- Pole **ID** (szerokość: 6 znaków), ma dopisywane wiodące zera, tak aby zajmowało całą szerokość kolumny.
- Pole **nick** (szerokość: 10 znaków) jest wyrównane do lewej.
- Pole **level** (szerokość: 7 znaków) jest wyrównane do prawej.
- Pole **pojemn** (szerokość: 6 znaków) też jest wyrównane do prawej i ma zawsze wypisaną jedną cyfrę po przecinku (dla liczb całkowitych jest to 0).

Przyjmij, że reprezentacja tekstowa liczb nie jest nigdy szersza, niż przyjęta szerokość kolumn (liczby nie są zbyt duże).

Uwaga 2: Zaprojektuj przynajmniej dwa własne manipulatory bezargumentowe do formatowania wartości poszczególnych pól.

Przykład raportu z serii wymyślonych wartości dla podanego typu danych:

Sat Sep 28 20:43:06 2015				
DELL-17				
+-----+-----+-----+-----+				
ID	nick	level	pkt	
+-----+-----+-----+-----+				
000004	Piasek	7	30.0	
000008	Piesek	12	100.9	
000012	Pasek	15	5432.1	
000016	Pisak	2	50.0	
+-----+-----+-----+-----+				
000000		15	5432.1	
+-----+-----+-----+-----+				