

LAB 13

Programowanie 2

Celem zadania jest stworzenie narzędzia do **prognozowania zaliczenia studentów** z pewnego **przedmiotu** oraz wykonania pewnych prostych **statystyk**.

Dane studentów oraz uzyskane przez nich **punkty** zapisane są w **pliku tekstowym**.

Każda linia w pliku zawiera *obowiązkowo* **imię** i **nazwisko** studenta, a następnie **listę uzyskanych punktów** (studenci mogą mieć różną liczbę punktów lub nie posiadać ich wcale). Przyznawane są tylko **całe punkty** (wartości całkowite) za zadanie (tj. 0, 1, 2, ...).

Zaliczenie uzyskuje się, wg skali zdefiniowanej w mapie **levels**. Student uzyskuje zaliczenie, gdy uzyska minimum punktów (patrz wartości zdefiniowane w mapie). Wystawiana **ocena** przyjmuje wartość całkowitą (tj. nie ma połówkowych ocen)

Funkcja **main** jest gotowa i nie wolno jej modyfikować poza komentowaniem lub odkomentowaniem fragmentów.

Szkielety klas są zdefiniowane i **nie** definiujemy żadnych dodatkowych metod.

UWAGA

Przed wykonaniem zadania, zapoznaj się z **szablonami algorytmów** w pliku **my_algorithm.h**. Wykorzystuj je w implementacjach klas. Zastanów się, co jest do nich **wymagane**? Wszystkie **kryteria** w tym zadaniu definiujemy za pomocą **lambda**.

ETAP 1

Lista studentów oraz ich punktów odczytana z pliku „dane.txt”

W klasie **student** zdefiniuj konstruktor oraz **operator<<** (wypisz **surname**, **name**, **mark**, a do wypisania punktów studenta zastosuj **my_for_each** oraz **lambda**)

W klasie **evaluation** zdefiniuj konstruktor, który odczyta dane z pliku tekstowego o nazwie przekazanej w parametrze oraz **operator<<**, który wypisze zawartość obiektu klasy **evaluation** (do przejrzenia listy studentów zastosuj **my_for_each** oraz **lambda** dla każdego studenta, gdzie wykorzystaj wypisywanie zdefiniowane dla pojedynczego studenta).

ETAP 2

Sortowanie leksykograficzne (*litery alfabetu angielskiego- metoda **sort***) danych studentów w klasie **evaluation** wg pola **surname**, a gdy są takie same decyduje pole **name**.

Zastosuj metodę **sort** dla listy studentów oraz zdefiniuj odpowiednią **lambda**.

ETAP 3

Wyznaczanie oceny studenta (mark) na podstawie sumy uzyskanych punktów oraz kryteriów wystawiania oceny z mapy levels

W klasie **student** zdefiniuj:

sum_of_points

zwraca sumę punktów; zastosuj algorytm **my_accumulate** oraz odpowiednie wyrażenie **lambda**

W klasie **evaluation** zdefiniuj metodę:

set_marks

metoda wypełnia pole **mark** dla listy kolejnych studentów (nie ma ocen połówkowych, tylko liczby całkowite 2, 3, 4, 5).

Do przeglądania listy studentów zastosuj **my_for_each**, do przeglądania mapy **levels** (z niej odczytaj kryteria oceny) zastosuj **my_find_if**. Zdefiniuj odpowiednie **lambda**.

W klasie **evaluation** zdefiniuj metodę:

reset_points

metoda dla kolejnych studentów przegląda ich punkty i do każdego zadania ocenionego na >0 dodaje podaną liczbę punktów.

Do przeglądania listy studentów zastosuj **my_for_each**, do modyfikacji listy punktów **my_transform**. Zdefiniuj odpowiednie **lambda**.

ETAP 4

Usunięcie studentów, którzy nie zaliczyli

W klasie **evaluation** zaimplementuj metodę

clear_not_passing

Dla listy studentów, zastosuj metodę **remove_if** oraz **lambda**.

ETAP 5

Histogram punktacji

Tworzony histogram określa, ile w sumie było rozwiązań zadań, które uzyskały konkretną liczbę punktów. Do przetrzymywania danych należy użyć **map<int,int>** gdzie kluczem jest liczba punktów (0, 1, 2, ..., **max**) a wartością liczba rozwiązań za tę liczbę punktów (zakres mapy **max** oznacza maksymalną liczbę punktów przyznanych za zadanie – obejrzyj wyniki).

W klasie **evaluation** zaimplementuj metodę **histogram**, która zwraca utworzoną mapę histogramu punktacji.

Metoda w pierwszej części dla każdego studenta wywoła metodę **histogram** z klasy **student**, gdzie parametrem jest wspólnie tworzona mapa. Wykorzystaj **my_for_each** oraz **lambda**.

W drugim wywołaniu **my_for_each** należy **wyświetlić** diagram „słupkowy” histogramu (patrz wyniki poniżej), gdzie w **lambda** zdefiniuj wypisywanie dla elementu mapy.

Wyniki

```
1: details from file -----
Pulewski      Jan           , ocena: 2: 7 3 8 5 6 10 6 3 9
Arkizot       Tadeusz      , ocena: 2: 2 2 4 7 4 3 1 2 5 7
Arioleta      Ewelina      , ocena: 2: 10 6 7 7 8 6 10 7
Magnucka      Emilia       , ocena: 2: 7 7 8 9 6 10 10 10 8 9
Torres        Zofia        , ocena: 2: 3 5 3 6 7 2 3 4 2 1
Molenda       Jolanta      , ocena: 2: 4 7 10 3 6
Kowalski      Adam         , ocena: 2: 10 10 11 8 10 10 10 10 10 10
Drugi         Adam         , ocena: 2:
Trzeci        Wojciech     , ocena: 2: 3 10 15
Ten           Adam         , ocena: 2: 10 12 12 8 10 10
```

2: sorted by surname, name -----

Arioleta	Ewelina	, ocena: 2: 10 6 7 7 8 6 10 7
Arkizot	Tadeusz	, ocena: 2: 2 2 4 7 4 3 1 2 5 7
Drugi	Adam	, ocena: 2:
Kowalski	Adam	, ocena: 2: 10 10 11 8 10 10 10 10 10
Magnucka	Emilia	, ocena: 2: 7 7 8 9 6 10 10 10 8 9
Molenda	Jolanta	, ocena: 2: 4 7 10 3 6
Pulewski	Jan	, ocena: 2: 7 3 8 5 6 10 6 3 9
Ten	Adam	, ocena: 2: 10 12 12 8 10 10
Torres	Zofia	, ocena: 2: 3 5 3 6 7 2 3 4 2 1
Trzeci	Wojciech	, ocena: 2: 3 10 15

3: marks -----

Arioleta	Ewelina	, ocena: 4: 12 8 9 9 10 8 12 9
Arkizot	Tadeusz	, ocena: 3: 4 4 6 9 6 5 3 4 7 9
Drugi	Adam	, ocena: 2:
Kowalski	Adam	, ocena: 5: 12 12 13 10 12 12 12 12 12 12
Magnucka	Emilia	, ocena: 5: 9 9 10 11 8 12 12 12 10 11
Molenda	Jolanta	, ocena: 2: 6 9 12 5 8
Pulewski	Jan	, ocena: 4: 9 5 10 7 8 12 8 5 11
Ten	Adam	, ocena: 4: 12 14 14 10 12 12
Torres	Zofia	, ocena: 3: 5 7 5 8 9 4 5 6 4 3
Trzeci	Wojciech	, ocena: 2: 5 12 17

4: passing students -----

Arioleta	Ewelina	, ocena: 4: 12 8 9 9 10 8 12 9
Arkizot	Tadeusz	, ocena: 3: 4 4 6 9 6 5 3 4 7 9
Kowalski	Adam	, ocena: 5: 12 12 13 10 12 12 12 12 12 12
Magnucka	Emilia	, ocena: 5: 9 9 10 11 8 12 12 12 10 11
Pulewski	Jan	, ocena: 4: 9 5 10 7 8 12 8 5 11
Ten	Adam	, ocena: 4: 12 14 14 10 12 12
Torres	Zofia	, ocena: 3: 5 7 5 8 9 4 5 6 4 3

5: histogram for points - ALL STUDENTS -----

```
pkt 3: **
pkt 4: *****
pkt 5: *****
pkt 6: ****
pkt 7: ***
pkt 8: *****
pkt 9: *****
pkt 10: *****
pkt 11: ***
pkt 12: *****
pkt 13: *
pkt 14: **
pkt 17: *
(3,2)(4,5)(5,8)(6,4)(7,3)(8,7)(9,10)(10,6)(11,3)(12,19)(13,1)(14,2)(17,1)
```

5: histogram for points - PASSING STUDENTS -----

```
pkt 3: **
pkt 4: *****
pkt 5: *****
pkt 6: ***
pkt 7: ***
pkt 8: *****
pkt 9: *****
pkt 10: *****
pkt 11: ***
pkt 12: *****
pkt 13: *
pkt 14: **
(3,2)(4,5)(5,6)(6,3)(7,3)(8,6)(9,9)(10,6)(11,3)(12,17)(13,1)(14,2)
```

Press any key to close this window . . .