

Języki Programowania, Zadanie 11

Napisz program w C++, który modeluje ocenianie studentów na różnych kursach.

1. Klasa bazowa: ScoreList

Przechowuje listę ocen dla jednego studenta lub kursu.

Wymagania:

Pola:

- double* scores – dynamicznie alokowana tablica ocen,
- int count – liczba ocen,
- std::string label – etykieta (np. nazwa studenta lub kursu).

Konstruktory:

1. Konstruktor przyjmujący: ○ etykietę (label), ○ liczbę ocen.
Ma alokować tablicę scores i wypełnić ją zerami.
2. Konstruktor kopiujący (głęboka kopia).
3. Destruktor zwalniający tablicę.

Metody:

- void FillRandom(int min, int max) – losuje oceny,
- void PrintLabel – wypisuje etykietę
- void Print() – wypisuje listę ocen.

2. Klasa pochodna: ExamResults

Przedstawia wyniki z konkretnego egzaminu.

Dodatkowe pola:

- std::string examiner – imię i nazwisko prowadzącego egzamin,
- double passingThreshold – próg zaliczenia.

Wymagania:

Konstruktor przyjmujący:

- etykietę (np. „Egzamin z matematyki”),
- liczbę ocen, • osobę prowadzącą,
- próg zaliczenia. Metody:
- void Print() – powinna: ○ wypisać nazwę kursu (label), ○ wypisać prowadzącego, ○ wypisać próg zaliczenia, ○ wywołać ScoreList::Print() aby pokazać oceny.

3. Funkcja main W main():

1. Stwórz obiekt ScoreList – np. lista ocen studenta A, 5 ocen.
2. Stwórz obiekt ExamResults – np. wyniki egzaminu z fizyki; prowadzący: Hanna Zbroszczyk, próg 51%.
3. Wylosuj oceny w obu obiektach.
4. Wyświetl informację o obu obiektach za pomocą odpowiednich metod.

4. Parametry wywołania programu

Na koniec w funkcji main należy utworzyć nowy kurs (typ. ScoreList), którego nazwa jest podana jako **pierwszy parametr wywołania programu**. Należy mu również ustawić liczbę ocen podaną jako **drugi parametr** wywołania programu (uwaga, program powinien wypisywać odpowiednie ostrzeżenie, jeśli liczba podanych parametrów jest nieprawidłowa!). Następnie informacje o kursie należy wypisać podobnie jak dla studenta A.

Parametry wywołania programu (zmiany w deklaracji funkcji main): int

```
main(int argc, char **argv){  
    }  
}
```

Gdzie: argc - liczba parametrów, z którymi został wywołany

program **argv - tablica parametrów (każdy jako char*):

argv[0] - nazwa programu

argv[1], argv[2]... - kolejne parametry wywołania programu

Np. wywołanie programu program mogło by mieć formę:

./program Ala 2 3 Wtedy:

//argc == 4

//argv[0] == "program"

//argv[1] == "Ala"

//argv[2] == "2"

//argv[3] == "3" || **zmiana "3" na 3 (liczbę) – funkcja atoi. np int a = atoi("3"); z biblioteki <stdlib.h>**

Przyporządkowanie tych parametrów wykonuje się samoistnie w momencie wywołania programu, jedynym wkładem programisty jest zadeklarowanie funkcji main w formie int main(int argc, char **argv)