

Podstawy programowania, 24.04.2017

Zadanie 6

Zadanie polega na stworzeniu bazy danych studentów Politechniki Warszawskiej. Załóżmy, że każdy student może być opisany przez następujący zestaw danych: `char imie[30]`, `char nazwisko[100]`, `int wiek`, `int rokStudiow`, `int ECTS`, `int wydzial`.

Do wykonania:

1) Stwórz strukturę `Student` zawierającą składniki: `char imie[30]`, `char nazwisko[100]`, `int wiek`, `int rokStudiow`, `int ECTS`, `int wydzial`. Umieść definicję struktury w pliku `Student.h`. Stwórz 2 funkcje powiązane ze strukturą `Student`: `void UstawStudent(Student *stud, char *Imie, char* Nazwisko, int Wiek, int RokStudiow, int Ects, int Wydzial)` oraz `void WypiszStudent(Student *stud)` ustawiające i wypisujące składniki struktury `Student`. Umieść deklaracje funkcji w pliku `Student.h` a ich definicje w pliku `Student.c`. Ustawiamy poleceniem `typedef` nazwę struktury na `Student`. (3 p.)

2) W funkcji głównej (`main`) tworzymy dynamicznie tablicę typu `Student`. Tablicę wypełniamy danymi z pliku `dane.txt`, w którym w pierwszej linijce przechowywana jest liczba studentów zawartych w pliku. Po wypełnieniu tablicy danymi z pliku wypisujemy studentów na ekran. (1 p.)

Całość kompilujemy w linii poleceń używając pliku `Makefile` oraz polecenia `make`. (1 p.)

Dodatkowo:

Zauważmy, że imię, nazwisko oraz wiek pełnią rolę danych osobowych (można je wykorzystać nie tylko do opisu studenta), zatem możemy je umieścić w jednej osobnej strukturze o nazwie `DaneOsobowe`. W strukturze `Student` zaś zamiast imienia, nazwiska i wieku tworzymy obiekt `DaneOsobowe`

Uwaga!

W plikach `.h` (gdzie mamy definicje struktur) używamy następujących instrukcji preprocesora:

```
#ifndef STUDENT_H
#define STUDENT_H
... (zawartość pliku)
#endif //STUDENT_H
```