Marcel Kawski

Dokumentacja

Projekt nr 3

Wstępne założenia:

* Program swoje działanie opiera na rozbudowanym Menu umożliwiającym w każdym momencie wybór opcji przez użytkownika.
* Program pobiera od użytkownika dane dotyczące zbioru książek. Są to kolejno:
* Tytuł (możliwe są wszystkie znaki graficzne i spacje, może się powtarzać, max 20 znaków)
* Kraj pierwszego wydania (możliwe są tylko litery i spacje, może się powtarzać, max 15 znaków)
* Numer (unikatowy dla każdej książki, składa się tylko z 5 cyfr).
* Program sprawdza poprawność wprowadzanych danych, zarówno poprawność danych dotyczących książek, jak i danych dotyczących poruszania się po MENU programu.
* W przypadku wprowadzenia złych danych program umożliwia ich ponowne wpisanie.
* Program umożliwia:
  + - tworzenie nowych katalogów
    - zmianę nazwy dowolnego katalogu
    - usunięcie istniejącego katalogu z zawartością
    - wpisywanie nowych rekordów opisywanych przez trzy dane
    - modyfikację już istniejących rekordów, w szczególności modyfikację jednej danej dla jednego rekordu, poprzez zastosowanie odpowiedniego menu umożliwiającego wybór odpowiedniej danej
    - przenoszenie rekordów z jednego katalogu do innego
    - wyszukiwanie i wyświetlanie listy rekordów wg zadanych parametrów z podaniem wszystkich danych dla każdego rekordu oraz katalogu w którym się znajduje, wyszukane rekordy wyświetlane są według kolejności alfabetycznej katalogów
    - wyświetlanie nazw wszystkich katalogów w kolejności alfabetycznej,
    - wyświetlanie zawartości dowolnie wybranego katalogu (pozycje w kolejności nazw książek) z podaniem wszystkich danych rekordów
    - wyjście z programu po uprzednim potwierdzeniu tej czynności przez użytkownika.
* Program automatycznie zmienia pierwszą literę nazwy katalogu, tytułu lub kraju na wielką.

Opis funkcji:

* **void dodajKatalog (struct Handler \*h)**

Funkcja odpowiada za dodawanie nowych katalogów (elementów listy głównej) i na bieżąco sprawdzanie poprawności wprowadzanych danych.

* **void dodajKsiazke (struct Handler h)**

Funkcja odpowiada za dodawanie nowych książek (elementów list podrzędnych) i na bieżąco sprawdzanie poprawności wprowadzanych danych.

* **void poprawDane (struct Handler \*h)**

Funkcja odpowiada za poprawianie wpisanych rekordów, dając wybór użytkownikowi wybór, który rekord chce poprawić i na bieżąco sprawdza poprawność nowych danych.

* **void przeniesRekord (struct Handler \*h)**

Funkcja odpowiada za przenoszenie książek do innych katalogów.

* **void sortujKatalogi (struct Handler \*h)**

Funkcja odpowiada za sortowanie alfabetyczne katalogów.

* **void usunKatalog (struct Handler \*h)**

Funkcja odpowiada za usuwanie katalogów wraz z zawartością.

* **void sortujKatalogi (struct Handler \*h)**

Funkcja odpowiada za sortowanie alfabetyczne katalogów.

* **void wypiszKatalogi (struct Handler \*h)**

Funkcja odpowiada za wypisanie katalogów.

* **void wypiszKsiazki (struct Handler \*h)**

Funkcja odpowiada za wypisanie książek wybranego przez użytkownika katalogu.

* **void wypiszWgParametrow (struct Handler h)**

Funkcja odpowiada za wypisywanie książek, której odpowiadają parametrom podanym przez użytkownika posortowanych wg kolejności alfabetycznej katalogów.

* **void zmienNazwe (struct Handler \*h)**

Funkcja umożliwia zmianę nazwy dowolnego katalogu.