

Opis wymagań dla programu „Biblioteka prac dyplomowych”

1 Podstawowe założenia

Napisać program do obsługi biblioteki prac dyplomowych, tj. licencjackich, inżynierskich oraz magisterskich.

Dane o pojedynczej pracy przechowywane powinny być w klasie *Praca*. Klasa ta powinna zawierać następujące składowe (w ostatnim nawiasie przy każdym polu podano typ danych dla pola):

- id – unikalny identyfikator pracy (int);
- typPracy – dopuszczalne wartości („inz”, „lic”, „mgr”) (string);
- tytuł – tytuł pracy, nie może być pusty (string);
- nazwiskoAutora – nie może być puste (string);
- imionaAutora – imiona autora oddzielone przecinkami (string);
- rok – rok powstania pracy (int, min. rok to 2000);
- nazwiskoPromotora – nazwisko promotora pracy (string);
- imionaPromotora – imiona promotora, oddzielone przecinkami (string);
- slowaKluczowe – słowa kluczowe, oddzielone średnikami (string, może być pusty);
- streszczenie – krótki opis słowny tematyki pracy (string, może być pusty, maks. dł. 1000 znaków);

Listą prac należących do biblioteki zarządzać powinna klasa *Biblioteka* zawierająca następujące składowe:

- prace – tablica (lista, wektor) obiektów typu *Praca* reprezentujących prace w bibliotece. Można założyć, że maksymalna liczba prac w bibliotece nie przekroczy 1000 egzemplarzy;
- ilePrac – ile prac jest obecnie w bibliotece (int).

1.1 Funkcjonalność programu

Funkcje programu powinny obejmować:

1. Wyświetlanie listy prac w formacie skróconym składającym się z identyfikatora, typu pracy, tytułu, inicjałów autora i jego nazwiska oraz roku powstania np.:
123 mgr A. Nowak „Algorytmy genetyczne w problemie TSP” 2008
2. Wyświetlanie listy prac w formacie pełnym zawierającym wszystkie dane o pracy. Poszczególne elementy opisu pracy powinny być wyświetlone w kolejnych wierszach, tzn.:

Id: 123

Typ: mgr

Autor: Adam Nowak

Tytuł: Algorytmy genetyczne w problemie TSP

Słowa kluczowe: algorytmy genetyczne; problem komiwojażera; TSP

Rok: 2008

Promotor: Jan Kowalski

Streszczenie: Praca porównuje wpływ wybranych metod kodowania osobników w klasycznym algorytmie genetycznym na jakość rozwiązań dla problemu TSP.

3. Dodawanie prac do biblioteki – użytkownik proszony jest o podanie wartości dla poszczególnych elementów dodawanej pracy;
4. Edycja danych o pracy (na wypadek błędów w opisie) – możliwość zmiany wartości wybranej przez użytkownika składowej opisu spośród: imion i nazwiska autora, typu, roku powstania, tytułu, roku wydania, słów kluczowych oraz opisu. Praca do edycji wybierana jest na podstawie podanego identyfikatora. Przed rozpoczęciem edycji powinny zostać wyświetlone informacje o edytowanej pracy.
5. Wyszukiwanie prac na podstawie wybranego kryterium:
 - nazwiska autora,
 - nazwiska promotora,
 - tytułu,
 - słów kluczowych,
 - streszczenia,
 - roku powstania.

Wyszukiwanie powinno uwzględniać częściowe dopasowanie, gdy użytkownik poda np. fragment nazwiska autora lub tytułu. Wyszukiwanie dla roku powinno być dokładne. Wyniki wyszukiwania powinny być wyświetlane w kolejności od najnowszych do najstarszych.

6. Funkcje dodatkowe z osobnym podmenu:
 - (a) Wyświetlanie liczby wszystkich prac, w tym również podziałem na poszczególne typy.
 - (b) Wyświetlanie liczby prac dla poszczególnych promotorów – posortowane alfabetycznie wg nazwiska promotora.
 - (c) Wyświetlanie 10 najczęściej powtarzających się słów kluczowych.

Dostęp do poszczególnych funkcji programu powinien być realizowany przez menu wyświetlane po uruchomieniu programu oraz po każdorazowym zakończeniu wykonywania wybranej wcześniej funkcji.

Każda z funkcjonalności programu powinna być implementowana przez odrębną metodę z klas Pracownik oraz Biblioteka. Dane przechowywane powinny być w pliku o nazwie „prace.dat” odczytywanym *automatycznie* po uruchomieniu programu. Jeżeli plik z danymi nie istnieje, to powinien zostać utworzony automatycznie (pusty). Po każdej modyfikacji danych należy je zapisać do wspomnianego pliku, tak aby były widoczne po ponownym uruchomieniu programu.

Format zapisu danych w pliku jest dowolny – z powodu kłopotliwej obsługi kodowania znaków w języku C/C++ można przyjąć, że dane będą wprowadzane i wyświetlane za pomocą kodowania cp1250 (do zmiany kodowania w konsoli systemu Windows służy polecenie „chcp 1250”).

Wprowadzane przez użytkownika dane powinny być weryfikowane i jeżeli będą błędne, to program powinien prosić o ich ponowne podanie.