

Zadanie 1

- (a) Zaimplementuj algorytmy realizujące podane poniżej problemy.
- (b) Problemy należy rozwiązać **zgodnie** z przedstawionymi w pliku drzewabinarne.pdf instrukcjami.
- (c) Komunikacja programu z użytkownikiem powinna się odbywać za pomocą plików o podanych poniżej formatach.

Problem 1 (drzewo AVL)

- (A)** Zaimplementuj operacje (search, insert, delete, KLP) na drzewie AVL.
- * Węzeł drzewa AVL powinien zawierać: klucz (typu int), współczynnik wyważenia (element ze zbioru {-1, 0, 1}), wskaźnik na lewe poddrzewo, wskaźnik na prawe poddrzewo.
 - * W węźle drzewa nie można umieszczać wskaźnika na ojca tego węzła.
 - * Po wstawieniu (usunięciu) elementu do drzewa AVL poprawiamy współczynniki wyważania jedynie na ścieżce wstawianego (usuwanego) elementu.
- Do testowania drzewa AVL przygotuj:
- (a) Wczytywanie elementów (typu int) z pliku InTest1.txt do drzewa AVL:
InTest1.txt
46 43 52 46 45 765 73 5 63 45 4 65 67 65 73 56 24 53 42 34 23 465 376 93 65 8
 - (b) Losowanie elementów (typu int) z ustalonego zakresu i wczytywanie ich do drzewa AVL (kolejność losowanych elementów powinna zostać zapisana w pliku OutTest2.txt).
 - (c) Wykonanie operacji w wyniku wyboru pozycji z następującego menu.
- * Zapisz elementy drzewa AVL (wraz z wagami umieszczonymi w nawiasach) do pliku OutTest3.txt w kolejności KLP.
 - * Podaj wagę i poziom (korzeń jest na poziomie 0) wybranego elementu drzewa AVL.
 - * Dodaj element do drzewa AVL.
 - * Usuń element z drzewa AVL.
 - * Wypisz elementy drzewa AVL.

(B) Użyj struktury dwóch drzew AVL połączonych wskaźnikowo do realizacji słownika angielsko-polskiego i polsko-angielskiego.

W jednym z drzew powinny występować wyrazy polskie ze wskaźnikami na odpowiednie węzły w drugim drzewie zawierającym słowa angielskie i odwrotnie (tzn. np. węzeł zawierający słowo „book” (umieszczony w drzewie AVL ze słowami angielskimi) powinien zawierać wskaźnik na węzeł zawierający słowo „książka” (umieszczony w drugim drzewie AVL ze słowami polskimi) i odwrotnie.

Zakładamy, że każdy wyraz (zarówno polski jak i angielski) może mieć tylko jedno tłumaczenie.

Aplikacja ma realizować operacje wstawiania, wyszukiwania i usuwania elementów słownika oraz powinna mieć możliwość wypisania całego słownika (słowo + tłumaczenie) w porządku KLP ze względu na słowa angielskie.

Program powinien umożliwiać wykonanie przedstawionych powyżej zadań realizowanych w wyniku wyboru opcji następującego menu:

Plik -> Wczytaj:

Wczytanie do słownika (niekoniecznie pustego) wyrazów (i ich tłumaczeń) z pliku tekstowego postaci:

InTest20501.txt

E1098 P3456 // słowo_angielskie1 tłumaczenie1

E4869 P3333 // słowo_angielskie2 tłumaczenie2

.....

Kolejne linie pliku nie muszą być posortowane ani ze względu na słowa polskie ani ze względu na słowa angielskie.

Plik -> Zapisz:

Zapisanie aktualnej zawartości obu drzew (słownika) do pliku tekstowego postaci przedstawionej powyżej. Słowa i ich tłumaczenia powinny być wpisywane do pliku w kolejności czytania z drzewa ze słowami angielskimi w porządku KLP.

Wstaw -> Słowo_polskie

Wstawienie do słownika nowego słowa polskiego wraz z jego tłumaczeniem (w przypadku, gdy dane słowo już istnieje należy podać odpowiedni komunikat)

Wstaw -> Słowo_angielskie

Wstawienie do słownika nowego słowa angielskiego wraz z jego tłumaczeniem (jak powyżej)

Wyszukaj -> Słowo_polskie

Znalezienie polskiego tłumaczenia dla słowa angielskiego (w przypadku braku słowa angielskiego w słowniku, należy podać odpowiedni komunikat)

Wyszukaj -> Słowo_angielskie

Znalezienie angielskiego tłumaczenia dla słowa polskiego (jak powyżej)

Usuń -> Słowo_polskie

Usunięcie ze słownika słowa polskiego wraz z tłumaczeniem (w przypadku jego braku, należy podać odpowiedni komunikat)

Usuń -> Słowo_angielskie

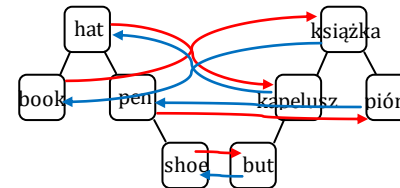
Usunięcie ze słownika słowa angielskiego wraz z tłumaczeniem (jak powyżej)

[Uwagi]

Należy stosować optymalne rozwiązania.

Przykład

book książka
pen pióro
hat kapelusz
shoe but



Problem 2 (drzewo BST)

(A) Zaimplementuj operacje (search, insert, delete, KLP) na drzewie BST.

- * Węzeł drzewa BST powinien zawierać: klucz (typu int), wskaźnik na lewe poddrzewo, wskaźnik na prawe poddrzewo.

- * W węźle drzewa nie można umieszczać wskaźnika na ojca tego węzła.

Do testowania drzewa BST przygotuj:

- (a) Wczytywanie elementów (typu int) z pliku InTest1.txt do drzewa BST:
InTest1.txt
46 43 52 46 45 765 73 5 63 45 4 65 67 65 73 56 24 53 42 34 23 465 376 93 65 8
 - (b) Losowanie elementów (typu int) z ustalonego zakresu i wczytywanie ich do drzewa BST (kolejność losowanych elementów powinna zostać zapisana w pliku OutTest2.txt).
 - (c) Wykonanie operacji w wyniku wyboru pozycji z następującego menu.
- * Zapisz elementy drzewa BST (wraz z wagami umieszczonymi w nawiasach) do pliku OutTest3.txt w kolejności KLP.
 - * Dodaj element do drzewa BST.
 - * Usuń element z drzewa BST.
 - * Wypisz elementy drzewa BST.

(B) Metoda Callana polega na powtarzaniu przygotowanych zdań sformułowanych na bazie wprowadzonych wcześniej wyrazów. Podręcznik do Callana składa się z pytań i odpowiedzi w języku angielskim, poprzedzonych „nowymi słówkami” wraz z tłumaczeniami. Autor książki jest nieco „roztargnionym” panem i zgubił plik z „nowymi” słówkami do 8 rozdziału. Spowodowało to zastój przy tworzeniu kolejnych rozdziałów, ponieważ nie było wiadomo, które słówka są „nowe” a które nie. Przyjaciel „roztargnionego” zlitował się nad kolegą i napisał program pomocny w konstrukcji podręcznika. Program składa się z dwóch części.

W pierwszym kroku, w celu odnalezienia „nowych” słówek, czytany jest tekst zawarty w pliku In0502.txt. Zadanie polega na sprawdzeniu, czy kolejne wyrazy znajdują się już w angielskim drzewie BST. Przy napotkaniu „nowego” wyrazu, program zapisuje go do pliku OutA0502.txt i na angielskim drzewie BST.

Następnie, w celu skonstruowania (polskiego) drzewa tłumaczeń, należy ręcznie uzupełnić w pliku OutA0502.txt polskie tłumaczenia dla zapisanych tam wyrazów angielskich.

W drugim kroku tworzone jest (polskie) drzewo tłumaczeń. Program powinien czytać kolejne wyrazy angielskie wraz z polskimi tłumaczeniami i generować polskie drzewo BST. Każdy nowo utworzony węzeł polskiego drzewa powinien być łączony wskaźnikowo z odpowiednim węzłem drzewa angielskiego.

Następnie należy przejrzeć angielskie drzewo BST w porządku KLP i wraz z polskimi tłumaczeniami odpowiednio zapisać do pliku OutB0502.txt.

Po uzupełnieniu obu drzew, program powinien mieć możliwość realizacji zadań wynikających z następującego menu

* wypisanie wyrazów angielskich (polskich) wraz z tłumaczeniami w kolejności KLP (do pliku)

* znalezienie wyrazu polskiego (angielskiego) wraz z tłumaczeniem

* dodanie nowego słowa wraz z tłumaczeniem

* usunięcie słowa wraz z tłumaczeniem

Przykład

In0502.txt

When do you think you will be ready to take the exam at the end of this book

I think I will be ready to take the exam at the end of this book in March

OutA0502.txt

when\'n\' do\'n\' you\'n\' think\'n\' will\'n\' be\'n\' ready\'n\' to\'n\' take\'n\' the\'n\' exam\'n\' at\'n\' end\'n\' of\'n\'

this\'n\' book\'n\' I\'n\' in\'n\' March

OutB0502.txt

ready gotowy\'n\' exam egzamin\'n\' do robić\'n\' be być\'n\' at w\'n\'

book książka\'n\' end koniec\'n\' in w\'n\' I ja\'n\' of z\'n\'

March marzec\'n\' think myśleć\'n\' take zdawać\'n\' the \'the\'n\' when kiedy\'n\'

to do\'n\' this ten\'n\' you ty\'n\' will wola

[Uwagi]

* Należy stosować optymalne rozwiązania.

* Dla ułatwienia zakładamy, że

* Wyrazy nie posiadają synonimów. W przypadku, gdy użytkownik wpisuje synonim, powinien pojawić się odpowiedni komunikat.

* Każdy zwrot (np. in order to, at the end of) złożony z kilku wyrazów jest tłumaczony jako zbiór oddzielnych słów.

* W czytanych tekstach nie ma znaków interpunkcyjnych. Wyrazy oddzielone są pojedynczą spacją.

* Wyrazy, dla których nie ma tłumaczeń (np. 'the') należy w polskim słowniku zapisać w apostrofach 'the'