Data Structure and Algorithm

Project 1

English-Chinese Dictionary based on Binary Search Tree

Contents

1. Introduction

Tree data structure is a widely used abstract data type in computer science, so it is necessary for you to better understanding it. In order to let you have a clearer understanding of tree data structure learned in the classes, this project designed for you is to use red-black tree and B+ tree to implement an English-Chinese dictionary. Because of the lexicographic order of English words, you are required to regard the English word as keys as well as use the Chinese word as values.

Considering you have not learned B+ tree in class, we provide you a learning material. And you can also learn more knowledge about it from website resources. Make sure you have fully understood red-black tree and B+ tree before coding.

2. Requirements

2.1 Basic Function

2.1.1 Initialization

Initialing red-black tree and B+ tree with several English-Chinese words which are randomly chosen from English dictionary. We organize these word in file file1_initial.txt, and the format of this file is, for example, like:

```
tree
树
data
```

```
数据
number
数字
……
```

The first line is an English word, and following is a Chinese word.

In order to provide convenience from downstream task, we choose words staring from different letters.

2.1.2 Preorder Print

Display the tree by PREORDER traversal on console.

Here we give you an example after inserting into trees with numbers 7,3,5,1,6. In this project, you are supposed to replace the number with words (English-Chinese word pair).

a) Print result for red-black tree

```
level=0 child=0 5(BLACK)
level=1 child=0 3(BLACK)
level=2 child=0 1(RED)
level=3 child=0 null
level=3 child=1 null
level=2 child=1 null
level=1 child=1 7(BLACK)
level=2 child=0 6(RED)
level=3 child=0 null
level=3 child=1 null
level=2 child=1 null
……
```

Hint: level means the level of a node, and child=0 means this node is a left child node, child=1 means this node is a right child node.

b) Print result for B+ tree

```
level=0 child=0 /5/
level=1 child=0 /1/3/
level=1 child=1 /5/6/7/
……
```

Hint: level means the level of a node, and child=0 means this node is the first child of its parent, and so on.

2.1.3 Insert/Delete

a). A batch of words which contain in a file .

The first line in the file represents which operation it will do, for example 'insert' or 'delete', and next lines are data. To learn more about the file format, you can see the sample files file2_delele.txt and file3_insert.txt we provided.

Each time you operate on 100 pieces of data, please call your PREORDER_PRINT method (in 2.1.2) to print the tree on the console if the total data size in tree is not larger than 500.

b). A single word

Hint: Before you insert or delete a word, you should check whether this word exists in the tree you have created.

2.1.4 Search

a). Some words in range and give their meanings.

For example, we give a query: please search from 'ant' to ‘cat’, then you give the words between ‘ant’ and ‘cat’ as well as their meanings. The boundary values don’t have to be exactly words.

b). A single word. Just give its Chinese meanings.

2.2 **Analysis Work**

Please call your program in the following ways:

1. First insert into trees the data in the file 1_initial.txt
2. Then delete the data in the file 2_delete.txt
3. Add the data in the file 3_insert.txt
4. Query a word
5. Query some words

For the first three steps, after each operation on 100 pieces of data, you should record the time used. For step 4 and step 5, just give the time totally consumed for each query.

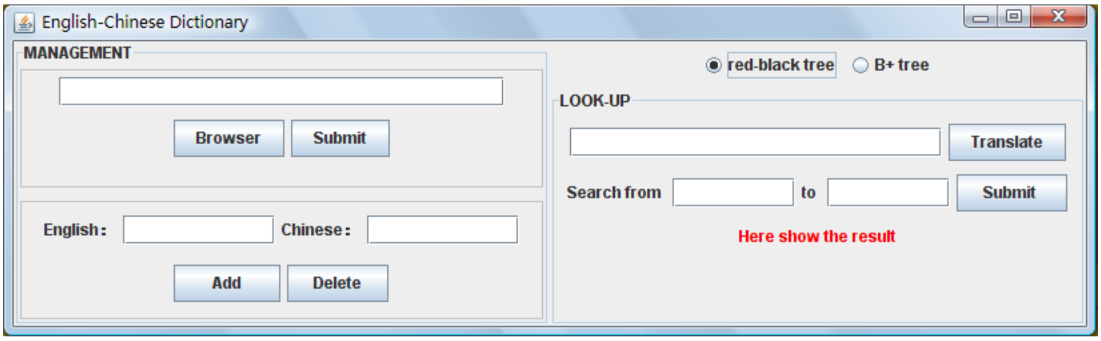
And finally give a document based on this analysis and testing time. You are recommended to compare the testing time and performance between two kinds of trees.

2.3 **User Interface(Optional)**

Note: this requirement of this part is not very strict. You are required to complete the task above, and then consider the user interface to make your program more useful.

To make your program more flexible and convenient, we don’t recommend that you complete all the task by command files. Instead, we prefer that you can provide a user interface through which users can import their files conveniently.

Here we provide a demo. And you're very welcome to show us a more user-friendly interface.


● red-black tree ○ B+ tree

Here show the result

Here are some Explanations for this demo user interface:

a) Left part: a management part.

This part gives two ways to extend or modify the dictionary. You can choose a file and execute the corresponding operations, or you can also add or delete a single word. If you just need to delete a word, only fill the English word.

b) Right top part: a choice part.

It means which kind of tree you are using to implement the dictionary. The user can make the choice.

c) Right down part: look-up part.

In this part, you are required to search words in the dictionary. You can Just give a translation for the word in the form, or you can also give the words and their translation results in the query range.

Note: Remind you that you should also print the tree forms on the console each time you operate on 100 pieces of data if the total data size in tree is not larger than 500.

3. Grading

Item	Cost	Description
General	50%	Correct implementation of basic function
Interview	25%	Correct answer face-to-face question
Documents	20%	Full and detail. Reasonable analysis
UI	5%	User-friendly and easy to use
Coding Style	5%	Proper comment
Bonus	5%	Creative thought.

4. Submissions

4.1 Documents

a) all the codes

b) a document based on the analysis and test time

4.2 Deadline

2018.10.14 23:59:59

Please submit to:

ftp://10.132.141.33/classes/17/181 数据结构与算法设计（郑骁庆）/WORK_UPLOAD/Project1

Naming: 学号_姓名.rar or .zip and so on.

4.3 Interview Time

2018.10.14 18:30 软件楼机房

5. Suggestions

a) Start early and make sure you have fully understood red-black tree and B+ tree before coding

b) You can start your program and test the correctness of your program with small batch English-Chinese words. After making sure the correctness of your program, you can use the file wo provided to test the running time, and then analysis you program.

c) If you have any questions, you can contact us with Wechat or in any other way. Good luck to you!