Politechnika Śląska
Wydział Informatyki, Elektroniki i Informatyki

Fundamentals of Computer Programming

« Red'n'Black»

author	Paweł Witkowski
instructor	dr inż. Pablo Ribalta Lorenzo
year	2020/2021
lab group	Friday, 14:30 – 16:00
deadline	2019-11-08

1 Project's topic

Write a program for sorting numbers in a set. The program executes commands stored in a file. A command is stored in a single line of the file. Available commands are:: `-i` input file name

Put precise description of your task – copy it from your instructor's task list.

Use a serif font for the main text. Justify the text (left and right alignment).

2 Analysis of the task

The task focuses on sorting numbers and order it in red black tree.

Analyse the task before you start to implement it! Describe data structures and algorithms.

2.1 Data structures

A red black tree is used to store values read from an input file all most like binary tree but it auto balans it self. Each node of a tree has 0, 1, or 2 children. Less values are stored in the left subtree, greater values – in the right subtree.

Describe data structures you use in your project (lists, trees, heaps, , . . .). Explain your choice. Add a figure.

2.2 Algorithms

The program add numbers to a red black tree and then prints numbers in depth-first search. Adding numbers and printing needs average time $O(logn)$. The pessimistic case (when tree degenerates to a list) not needs longer time: $O(logn)$ [**?**].

Describe applied algorithms and explain your choice.

Use mathematical mode for math formulae, eg: $O(logn)$.

If you use some literature, add references (like [?]).

3 External specification

This is a command line program. The program requires names of input. Put input file name after `-i` switch eg:

```
program -i input-file
```

File is a text file. The program called with no parameters prints help (a short manual).

Program call

```
program
```

prints a short manual. Program called with incorrect parameters prints an error message:

```
Incorrect paramters!
```

and prints help.

External specification is a short manual for a user. You should specify here how to use your program, number of parameters, file format of input and output data, . . .

For input commands and output messages use a proportional typeface.

4 Internal specification

This part is a technical documentation for programmers.

The program is implemented with structural paradigm. User interface is separated from program's logic.

4.1 Program overview

Program overview aims at providing a reader with a synthetic description of internal flow.

The main function verifies parameters of the program. If the verification is negative, an appropriate message is printed. In case of positive verification, data are read with redAndBlack function. The function opens a file stream, reads commads and numbers from the stream, and sort numbers and insert it in red black tree. Finally the program prints numbers into an output file and remove delete element from red black tree.

4.2 Description of types and functions

Description of types and functions is moved to the appendix.

5 Conclusions

The program implements a simple sorting algorithm with a red black tree.

Appendix
Description of types and functions

RedAndBlack

Generated by Doxygen 1.8.20

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

Chapter 2

File Documentation

2.1 help.txt File Reference

Functions

- i (with file name) start program with input file 'add ' for adding numbers to a set
- eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output print (with a file name) prints sorted numbers in the file
- eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black tree of numbers (with indented levels) to the standard output. Numbers in red nodes are printed in round brackets
- eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black tree of eg (13)
- eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black tree of numbers in black nodes – in square graph (with a file name) – the same as graph

Variables

- eg add remove for removing number from a set
- eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black tree of numbers in black nodes – in square brackets

2.1.1 Function Documentation

2.1.1.1 eg()

```
eg add remove for removing number from a eg remove print for printing sorted numbers to the
standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black
tree of eg (
            13  )
```

2.1.1.2 graph()

eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black tree of numbers in black nodes - in square but a tree is printed into the file graph (
with a file *name*)

2.1.1.3 i()

i (
with file *name*)

2.1.1.4 numbers()

eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black tree of numbers (
with indented *levels*)

2.1.1.5 print()

eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers print (
with a file *name*)

2.1.2 Variable Documentation

2.1.2.1 brackets

eg add remove for removing number from a eg remove print for printing sorted numbers to the standard output eg print my sorted numbers eg print my sorted numbers graph prints a red black tree of numbers in black nodes - in square brackets

2.1.2.2 set

eg add remove for removing number from a set

2.2 InputFile.txt File Reference

2.3 RedAndBlack.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <cstdlib>
#include <vector>
#include <string>
#include "RedAndBlack.h"
```

Functions

- int main (int argc, char const *argv[])

2.3.1 Function Documentation

2.3.1.1 main()

```
int main (
            int argc,
            char const * argv[] )
```

2.4 RedAndBlack.h File Reference

```
#include <iostream>
#include <fstream>
#include <cstdlib>
#include <vector>
#include <string>
#include "RedAndBlack.h"
```

Functions

- void help ()
- vector< string > getLine (string str)
- vector< double > sortingNumbers (vector< double > tmpVector)
- vector< double > removeNumber (vector< double > tmpVector, double number)
- void printToFile (string name, bool plus, vector< double > tmpNumbers)
- void print (vector< double > tmpNumbers)
- void error (string command)
- void graphToFile (string name, bool plus, vector< double > tmpNumbers)
- void graph (vector< double > tmpNumbers)
- void redAndBlack (string inputFile)

2.4.1 Function Documentation

2.4.1.1 error()

```
void error (
            string command )
```

function to print error of command after show error show all allowed commands

Parameters

command	wrong command get from file

2.4.1.2 getLine()

```
vector<string> getLine (
            string str )
```

function to delete all spaces from string

Parameters

str	input string with spaces

Returns

vector with only words

2.4.1.3 graph()

```
void graph (
            vector< double > tmpNumbers )
```

function to print gparh in standart output

Parameters

tmpVector	vector of sorted numbers to print

2.4.1.4 graphToFile()

```
void graphToFile (
            string name,
            bool plus,
            vector< double > tmpNumbers )
```

function to print graph to new file or add at the end of file

Parameters

name	input name of file to output
plus	input element says if print to new file or add at the end of file
tmpVector	vector of sorted numbers to print

2.4.1.5 help()

```
void help ( )
```

function to show all commands when wrong one will be used

2.4.1.6 print()

```
void print (
            vector< double > tmpNumbers )
```

function to print sorted vector in standart output

Parameters

tmpVector	vector of sorted numbers to print

2.4.1.7 printToFile()

```
void printToFile (
            string name,
            bool plus,
            vector< double > tmpNumbers )
```

function to print sorted vector to new file or add at the end of file

Parameters

name	input name of file to output
plus	input element says if print to new file or add at the end of file
tmpVector	vector of sorted numbers to print

2.4.1.8 redAndBlack()

```
void redAndBlack (
            string inputFile )
```

Main function of program to choose what to do with inputed file Work as swich but on strings most of function is sequance of if At begin read "InputFile.txt" to know used commands Do the commands in order from first to last

Parameters

inputFile	name of input file

Get data from input file add it to 2D vector to know the order of comand

Start of swich

2.4.1.9 removeNumber()

```
vector<double> removeNumber (
            vector< double > tmpVector,
            double number )
```

function to delete chosen number only if it is in vector, change number to last and delete last element

Parameters

tmpVector	input vector with sorted numbers
number	element to delete from vector

Returns

vector with out deleted number and sorted in ASC order

2.4.1.10 sortingNumbers()

```
vector<double> sortingNumbers (
            vector< double > tmpVector )
```

function to sort vector of numbers in order by ASC

Parameters

tmpVector	input vector with not sorted numbers

Returns

vector with sorted numbers

Index