УО «Белорусский государственный университет информатики и радиоэлектроники»

Кафедра ПОИТ

Отчет по лабораторной работе №3.2

по предмету

Основы алгоритмизации и программирования

Вариант 14

Выполнил:

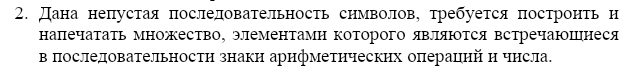
Крутько А.А.

Проверила:

Данилова Г.В.

Группа 251004

Минск 2022

Задание:  


Код программы на **Delphi**:

Program LabSecondBlockThird;

Uses

System.SysUtils;

Type

TSet = Set of Char;

Function ChooseOption(): String;

Var

Option: String;

IsRight: Boolean;

Begin

Writeln('Введите console, если хотите использовать консоль, file, если файл');

Repeat

IsRight := True;

Readln(Option);

Option := LowerCase(Option);

If (Option <> 'console') and (Option <> 'file') then

Begin

IsRight := False;

Writeln('Повторите ввод!');

End;

Until IsRight;

ChooseOption := Option;

End;

Function InputStrConsole(): String;

Var

Str: String;

IsIncorrect: Boolean;

Begin

Repeat

IsIncorrect := True;

Readln(Str);

If (Length(Str) = 0) Then

Begin

IsIncorrect := False;

Writeln('Повторите ввод!');

End;

Until IsIncorrect;

InputStrConsole := Str;

End;

Function TakeFileWay(): String;

Var

Way: String;

F: TextFile;

N: Integer;

IsCorrect: Boolean;

Begin

Repeat

IsCorrect := True;

Writeln('Введите путь к файлу');

Readln(Way);

If Not(FileExists(Way)) or Not(ExtractFileExt(Way) = '.txt') Then

Begin

Writeln('Файла в заданном пути нет или неправильное разрешение   
 файла');

IsCorrect := False;

End;

Until IsCorrect;

TakeFileWay := Way;

End;

Function IsCorrectStrFromFile(Var Str: String; Var F: TextFile): Boolean;

Var

IsCorrect: Boolean;

Begin

IsCorrect := True;

If Not Eof(F) Then

Begin

Readln(F, Str);

If (Length(Str) = 0) Then

Begin

Writeln('Проверьте правильность введенной строки');

IsCorrect := False;

End;

If (IsCorrect) and Not Eof(F) Then

Begin

Writeln('Уберите лишние данные');

IsCorrect := False;

End;

End

Else

Begin

Writeln('Нет данных для строки');

IsCorrect := False;

End;

IsCorrectStrFromFile := IsCorrect;

End;

Function InputStrFromFile(): String;

Var

Str, Way: String;

F: TextFile;

IsCorrect: Boolean;

Begin

Repeat

IsCorrect := True;

Way := TakeFileWay();

AssignFile(F, Way);

Try

Try

Reset(F);

IsCorrect := IsCorrectStrFromFile(Str, F);

Finally

Close(F);

End;

Except

IsCorrect := False;

Writeln('Файл нельзя прочитать');

End;

Until IsCorrect;

InputStrFromFile := Str;

End;

Function FindAnswer(Str: String): TSet;

Var

MySet, FinalSet: TSet;

I: Integer;

Begin

FinalSet := [];

MySet := ['('..'>'];

For I := 1 to Length(Str) do

Begin

If AnsiChar(Str[I]) In MySet Then

Begin

Include(FinalSet, AnsiChar(Str[I]));

Exclude(MySet, AnsiChar(Str[I]));

End;

End;

FindAnswer := FinalSet;

End;

Function TakeFinalInformation(Input: String): TSet;

Var

Str: String;

FinalSet: TSet;

Begin

If (Input = 'console') Then

Str := InputStrConsole()

Else

Str := InputStrFromFile();

FinalSet := FindAnswer(Str);

TakeFinalInformation := FinalSet;

End;

Procedure PrintInFile(FinalSet: TSet);

Var

F: TextFile;

Way: String;

IsCorrect: Boolean;

I: Integer;

Begin

Repeat

Way := takeFileWay();

AssignFile(F, Way);

Try

Try

ReWrite(F);

If FinalSet = [] Then

Writeln(F,'Нет нужных символов')

Else

Begin

Writeln(F,'Ваши символы в введенной строке: ');

For I := 0 to 255 do

If Chr(I) In FinalSet Then

Write(F,Chr(I):4);

End;

Finally

Close(F);

End;

Except

IsCorrect := False;

Writeln('В Файл нельзя вывести');

End;

Until IsCorrect;

End;

Procedure OutputAnswer(Output: String; FinalSet: TSet);

Var

Way: String;

I: Integer;

Begin

If (Output = 'console') Then

Begin

If FinalSet = [] Then

Writeln('Нет нужных символов')

Else

Begin

Writeln('Ваши символы в введенной строке: ');

For I := 0 to 255 do

If Chr(I) In FinalSet Then

Write(Chr(I):4);

End;

End

Else

PrintInFile(FinalSet);

End;

Var

Input, Output: String;

FinalSet: TSet;

Begin

Input := ChooseOption();

FinalSet := TakeFinalInformation(Input);

Output := ChooseOption();

OutputAnswer(Output, FinalSet);

Readln;

End.

Код программы на **C++**:

#include <iostream>

#include <set>

#include <string>

#include <fstream>

#include <algorithm>

using namespace std;

string chooseOption()

{

bool isIncorrect;

string input;

cout << "Введите console, если хотите использовать консоль, file, если файл" <<   
 endl;

do

{

isIncorrect = false;

cin >> input;

transform(input.begin(), input.end(), input.begin(), ::tolower);

if (input != "console" && input != "file")

{

cerr << "Повторите ввод" << endl;

isIncorrect = true;

}

} while (isIncorrect);

return input;

}

string inputStrConsole()

{

string str;

bool isIncorrect;

do

{

isIncorrect = false;

cin >> str;

if (str.empty() || str[0] == ' ')

{

isIncorrect = true;

cerr << "Повторите ввод" << endl;

}

} while (isIncorrect);

return str;

}

set<char> findAnswer(string str)

{

set<char> finalSet, mySet{'0','1','2','3','4','5','6','7','8','9','+','\*',':','-  
 ', '<', '>', '=', '(', ')' };

for (int i = 0; i < str.length(); i++)

if (mySet.find(str[i]) != mySet.end())

finalSet.insert(str[i]);

/\*tempSet.insert(str[i]);

set\_intersection(mySet.begin(), mySet.end(), tempSet.begin(), tempSet.end(), inserter(finalSet, finalSet.begin()));\*/

return finalSet;

}

bool checkExtension(string path)

{

bool isIncorrect = true;

if ((path[path.length() - 1] == 't') && (path[path.length() - 2] == 'x') &&   
 (path[path.length() - 3] == 't') && (path[path.length() - 4] == '.'))

isIncorrect = false;

return isIncorrect;

}

string takeFileWay()

{

string way;

bool isIncorrect;

do

{

isIncorrect = false;

cout << "Введите путь к файлу" << endl;

cin >> way;

fstream fin(way);

if (checkExtension || !fin.is\_open())

{

cout << "Проверьте правильность или нахождение вашего файла по заданному   
 пути" << endl;

isIncorrect = true;

}

fin.close();

} while (isIncorrect);

return way;

}

string takeStrFromFile(string way)

{

string str;

bool isIncorrect;

ifstream fin(way);

do {

isIncorrect = false;

if (!fin.eof())

{

getline(fin, str);

}

else

{

cerr << "В файле нет данных для строки" << endl;

isIncorrect = true;

way = takeFileWay();

}

if (!isIncorrect && !fin.eof())

{

cerr << "Уберите лишние данные" << endl;

isIncorrect = true;

way = takeFileWay();

}

} while (isIncorrect);

fin.close();

return str;

}

set<char> takeFinalInformation(string input)

{

set<char> finalSet;

string str, way;

if (input == "console")

{

str = inputStrConsole();

}

else

{

way = takeFileWay();

str = takeStrFromFile(way);

}

finalSet = findAnswer(str);

return finalSet;

}

void outputAnswerInFile(string way, set<char> finalSet)

{

bool isIncorrect;

ofstream fout(way);

do

{

isIncorrect = false;

if (finalSet.empty())

fout << "Нет нужных символов";

else

for (auto st : finalSet)

fout << st << '\t';

} while (isIncorrect);

fout.close();

}

void outputFinalInformation(set<char> finalSet, string outputChoice)

{

string way;

if (outputChoice == "console")

{

if (finalSet.empty())

cout << "Нет нужных символов";

else

for (auto i = finalSet.begin(); i != finalSet.end(); i++)

cout << \*i << '\t'; //auto st : finalSet cout << st << '\t'

}

else

{

way = takeFileWay();

outputAnswerInFile(way, finalSet);

}

}

int main()

{

setlocale(LC\_ALL, "RUSSIAN");

set<char> finalSet;

string inputChoice, outputChoice;

cout << "Данная программа выводит множество арифметических знаков и цифр" <<   
 endl;

inputChoice = chooseOption();

finalSet = takeFinalInformation(inputChoice);

outputChoice = chooseOption();

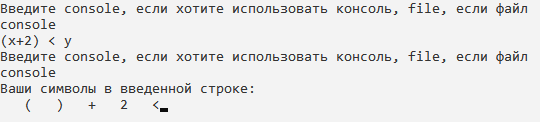
outputFinalInformation(finalSet, outputChoice);

}

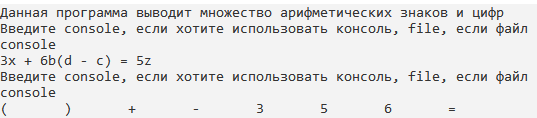
Код программы на **Java**:

import java.util.Scanner;  
import java.io.\*;  
import java.util.HashSet;  
import java.util.Set;  
public class LabSecondBlockThird {  
 static Scanner scan = new Scanner(System.in);  
  
 public static String chooseOption() {  
 boolean isIncorrect;  
 String input;  
 System.out.println("Введите console, если хотите использовать консоль, file,   
 если файл");  
 do {  
 isIncorrect = false;  
 input = scan.nextLine();  
 if (!input.equalsIgnoreCase("console") &&   
 !input.equalsIgnoreCase("file")){  
 System.err.println("Повторите ввод: ");  
 isIncorrect = true;  
 }  
 } while (isIncorrect);  
 return input;  
 }  
  
 public static String inputStrConsole() {  
 String str;  
 boolean isIncorrect;  
 do {  
 str = scan.nextLine();  
 isIncorrect = (str.isEmpty() || str.charAt(0) == ' ');  
 } while (isIncorrect);  
 return str;  
 }  
  
 public static HashSet <Character> findAnswer(String str) {  
 HashSet<Character> finalSet = new HashSet<>();  
 HashSet<Character> mySet = new HashSet<>(Set.of('0', '1', '2', '3', '4', '5',   
 '6', '7', '8', '9', '+', '-', '=', ':', '<', '>', '(', ')'));  
 for (int i = 0; i < str.length(); i++)  
 if (mySet.contains(str.charAt(i))) {  
 finalSet.add(str.charAt(i));  
 mySet.remove(str.charAt(i));  
 }  
 return finalSet;  
 }  
  
 public static String takeFileWay() {  
 String way;  
 boolean isIncorrect;  
 do {  
 isIncorrect = false;  
 System.out.println("Введите путь к файлу");  
 way = scan.nextLine();  
 File file = new File(way);  
 if (!file.exists() | !way.endsWith(".txt") | file.isDirectory()) {  
 isIncorrect = true;  
 System.err.println("Проверьте параметры файла");  
 }  
 } while (isIncorrect);  
 return way;  
 }  
  
 public static String inputStringFromFile() {  
 String str = "", way;  
 boolean isIncorrect;  
 do {  
 way = takeFileWay();  
 try (BufferedReader br = new BufferedReader(new FileReader(way))) {  
 if ((str = br.readLine()) != null) {  
 isIncorrect = (str.isEmpty() || str.charAt(0) == ' ');  
 } else {  
 isIncorrect = true;  
 System.err.println("В файле нет данных для строки");  
 }  
 if (!isIncorrect & br.readLine() != null) {  
 isIncorrect = true;  
 System.err.println("Уберите лишние данные");  
 }  
 } catch (IOException e) {  
 System.err.println(e.getMessage());  
 isIncorrect = true;  
 }  
 } while (isIncorrect);  
 return str;  
 }  
  
 public static void writeAnswerInFile(HashSet<Character> finalSet) {  
 boolean isIncorrect;  
 String way;  
 do {  
 way = takeFileWay();  
 isIncorrect = false;  
 try (PrintWriter pw = new PrintWriter(new FileWriter(way, false))) {  
 if (finalSet.isEmpty())  
 pw.write("Нет нужных символов");  
 else  
 pw.write(String.valueOf(finalSet));  
 } catch (IOException e) {  
 System.err.println(e.getMessage());  
 isIncorrect = true;  
 }  
 } while (isIncorrect);  
 }  
  
 public static HashSet<Character> takeFinalInformation(String input) {  
 String str;  
 HashSet<Character> finalSet;  
 str = (input.equalsIgnoreCase("console")) ? inputStrConsole() : inputStringFromFile();  
 finalSet = findAnswer(str);  
 return finalSet;  
 }  
 public static void outputAnswer(String outputChoice, HashSet<Character> finalSet) {  
 if (outputChoice.equalsIgnoreCase("console")) {  
 if (finalSet.isEmpty())  
 System.out.println("Нет нужных символов");  
 else  
 System.out.println(finalSet);  
 } else {  
 writeAnswerInFile(finalSet);  
 }  
 }  
  
 public static void main(String[] args) {  
 String inputChoice, outputChoice;  
 HashSet<Character> finalSet;  
 inputChoice = chooseOption();  
 finalSet = takeFinalInformation(inputChoice);  
 outputChoice = chooseOption();  
 outputAnswer(outputChoice, finalSet);  
 }  
}

Результат на **Delphi**:



Результат на **C++**:

****

Результат на **Java**:

