Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по лабораторной работе №10

Тема работы: Текстовые файлы

Выполнил

студент: гр. 851004 Никитко А.А.

Проверил: Фадеева Е.П.

Минск 2019

Содержание

[1 Постановка задачи 3](#_Toc10631288)

[2 Описание алгоритмов решения задачи 4](#_Toc10631289)

[3 Структура данных 5](#_Toc10631290)

[3.1 Структура данных алгоритма findStr 5](#_Toc10631291)

[3.2 Структура данных алгоритма assignF 5](#_Toc10631292)

[3.3 Структура данных алгоритма ScanFile 5](#_Toc10631293)

[4 Схема алгоритма решения задачи по ГОСТ 19.701-90 6](#_Toc10631294)

[4.1 Схема алгоритма решения задачи 6](#_Toc10631295)

[4.2 Схема алгоритма findStr 6](#_Toc10631296)

[4.3 Схема алгоритма fassignF 7](#_Toc10631297)

[4.4 Схема алгоритма ScanFile 8](#_Toc10631298)

[Приложение А 9](#_Toc10631299)

[Приложение B 11](#_Toc10631300)

# Постановка задачи

Дан текстовый файл F. Напечатать первую из самых коротких его строк.

# 2 Описание алгоритмов решения задачи

Таблица – Описание алгоритмов решения задачи

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  п.п. | Наименование алгоритма | Назначение алгоритма | Формальные параметры | Предполагаемый тип реализации |
| 1 | findStr  (minlen, stroka, result) | Определяет минимальную строку | minlen – возвращаемый параметр-переменная. Получает параметр-адрес без защиты;  stroka – возвращаемый параметр-переменная. Получает параметр-адрес;  result – возвращаемое значение функции. | Функция (result –  возвращаемое значение функции) |
| 2 | assignF  (f) | Открывает, если существует, файл или создает и открывает, если не существует. | f – возвращаемый параметр-переменная. Получает параметр-адрес без защиты. | Процедура |
| 3 | ScanFile (f, result) | Проходит по файлу f построчно и ищет первую из самых коротких строк. Записывает эту строку в результат. | f – возвращаемый параметр-переменная. Получает параметр-адрес без защиты;  result - возвращаемое значение функции. | Функция (result –  возвращаемое значение функции) |

# Структура данных

## Структура данных алгоритма findStr

Таблица 2 – Структура данных алгоритма findStr (minlen, stroka)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| minlen | integer | Минимальная длина | Формальный |
| stroka | string | Строка, длину которой сравниваем | Формальный |

## Структура данных алгоритма assignF

Таблица 3 – Структура данных алгоритма assignF (f)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| f | TextFile | Переменная текстового файла, которая связана с внешним текстовым файлом. | Формальный |
| filename | TextFile | Внешний текстовый файл | Локальная переменная |

## Структура данных алгоритма ScanFile

Таблица 4 – Структура данных алгоритма ScanFile (f, result)

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Тип параметра |
| f | TextFile | Переменная текстового файла, которая связана с внешним текстовым файлом. | Формальный |
| s | string | Строка, загружаемая из файла | Локальная переменная |
| str | string | Минимальная строка | Локальная переменная |
| len | integer | Длина минимальной строки | Локальная переменная |

# Схема алгоритма решения задачи по ГОСТ 19.701-90

## Схема алгоритма решения задачи

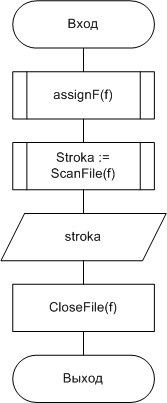


Рисунок – Схема алгоритма решения задачи

## Схема алгоритма findStr

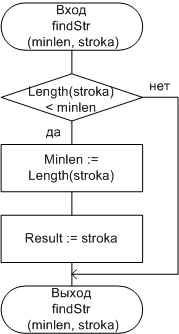


Рисунок 2– Схема алгоритма findStr

## Схема алгоритма fassignF

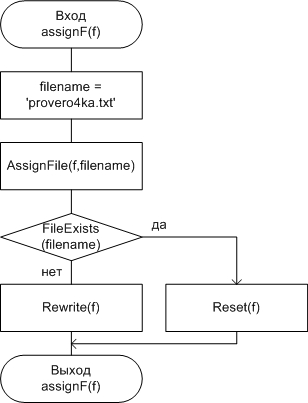


Рисунок 3 – Схема алгоритма assignF

## Схема алгоритма ScanFile

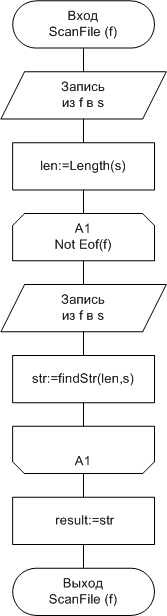


Рисунок 4 – Схема алгоритма ScanFile

Приложение А

(обязательное)

Исходный код программы

Program Laba10;

//Дан файл F. Напечатать первую из самых коротких его

//строк

//Подключение консольного приложения

{$APPTYPE CONSOLE}

//Подключаемые модули

Uses

SysUtils;

//Объявление переменных

var

f: TextFile;

stroka: string;

//f – текстовый файл

//str – минимальная строка

//Функция, определяющая короткую строку

function findStr (var minlen : Integer;

var stroka : string) : string;

begin

if Length(stroka) < minlen then

begin

minlen:=Length(stroka);

result:=stroka;

end;

end;

//Процедура открытия файла

procedure assignF(var f : TextFile);

const

filename = 'provero4ka.txt';

begin

AssignFile(f,filename);

if FileExists(filename) then

Reset(f)

else

Rewrite(f);

end;

//Функция прохождения по строкам файла и поиска самой

//короткой

function ScanFile (var f : TextFile) : string;

var

s, str: string;

len : Integer;

begin

Readln(f,s);

len:=Length(s);

while not (Eof(f)) do

begin

Readln(f,s);

str:=findStr(len,s);

end;

result:=str;

end;

begin

//Открытие файла

assignF(f);

//Определение первой самой короткой строки в файле

stroka:=ScanFile(f);

//Вывод результатов и закрытие файла

writeln('The shortest string of file : ',stroka);

CloseFile(f);

Readln;

end.

Приложение B

(обязательное)

Тест 1

Тестовая ситуация: 1 короткая строка

Исходные данные: текстовый файл

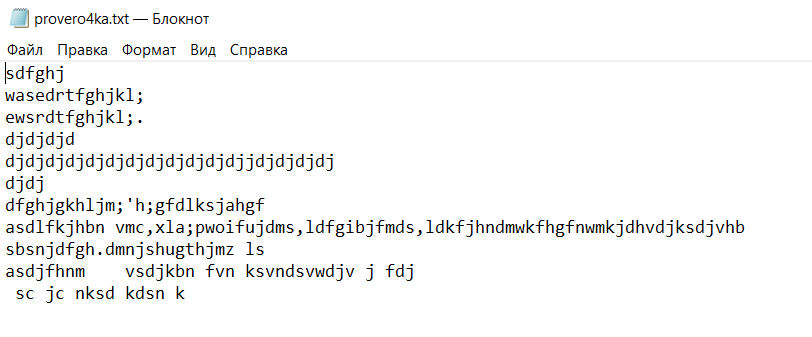


Рисунок 5 – Исходный текстовый файл

Ожидаемый результат: djdj

Полученный результат:



Рисунок 6 – Результат теста 1

Тест 2

Тестовая ситуация: несколько коротких строк, вывести первую из них

Исходные данные: текстовый файл

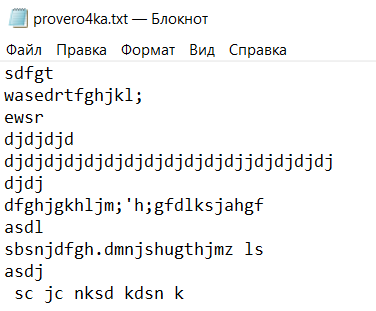


Рисунок 7 – Исходный текстовый файл

Ожидаемый результат: ewsr

Полученный результат:



Рисунок 8 – Результат теста 2

Тест 2

Тестовая ситуация: проверка работы приложения, при отсутствии текста в файле

Исходные данные: текстовый файл

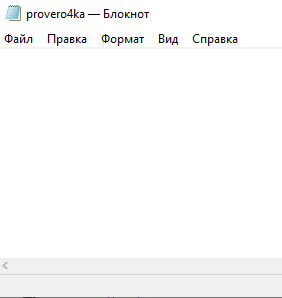


Рисунок 8– Исходный текстовый файл

Ожидаемый результат:



Полученный результат:



Рисунок 9 – Результат теста 2