

МАГИСТЕРСКАЯ ПРОГРАММА 09.04.01/12 Интеллектуальный анализ больших данных в системах поддержки принятия решений.

Вариант № 8

Дисциплина: Интеллектуальные технологии и системы

(И.О. Фамилия)

(И.О. Фамилия)

Москва, 2023

Цель: приобретение навыков проектирования и реализации основных элементов систем искусственного интеллекта

Задание: реализовать следующие функции:

- выбор текстового файла с помощью меню в текущем каталоге;
- редактирование выбранного файла;
- определять входит ли данное слово в выбранный файл.

1 Структурная схема и структурная карта Константайна

Программа, решающая поставленную задачу, состоит из нескольких модулей. Её структурная схема представлена на рисунке 1.



Рисунок 1. Структурная схема

Для демонстрации структуры вызова микромодулей на рисунке 2 представлена структурная карта Константайна.

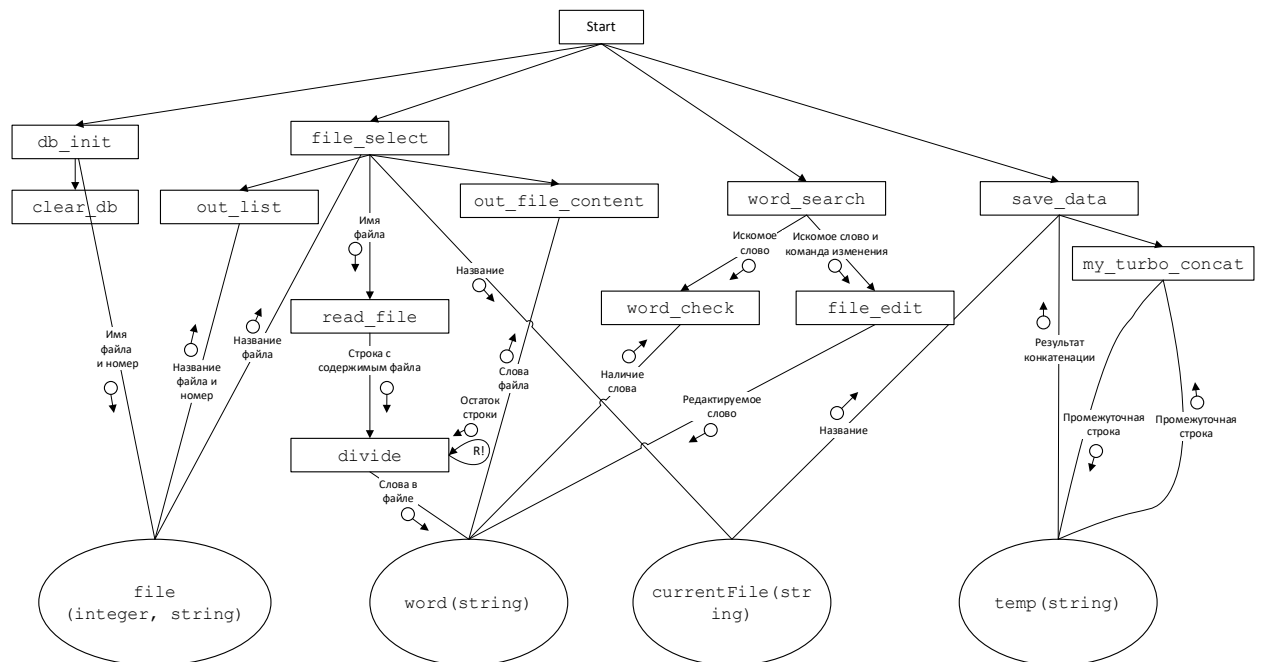


Рисунок 2. Структурная карта Константайна

2 Программа и её результат работы

CONSTANTS

```
file1="test1.txt"
file2="test2.txt"
file3="test3.txt"
file4="test4.txt"
```

DATABASE

```
file(integer, string)
word(string)
currentFile(string)
temp(string)
```

Predicates

```
start db_init clear_db file_select out_list read_file(string)
divide(string)
out_file_content word_search word_check(string)
file_edit(string, string)
save_data my_turbo_concat
```

Goal

```
start.
```

Clauses

```
% main branch
Start:-
    db_init,
    file_select,
    word_search,
    !, save_data.

% -----

% files list initialisation
db_init:-
    clear_db,
    assert( file(1, file1) ),
    assert( file(2, file2) ),
    assert( file(3, file3) ),
    assert( file(4, file4) ).

clear_db:-
    retract(file(_, _)), retract(word(_)),
    retract(currentFile(_)), retract(temp(_)), fail.
clear_db:- !.

% -----

% user file selection
file_select:-
    makewindow(1, 11, 3, "FILE SELECT", 0,0,25,80),
```

```

        write("Files loaded. Select file"), nl,
        out_list,
        readint(Pos),
        file(Pos,Name),
        read_file(Name),
        write("File loaded:"),
        assert( currentFile(Name) ),
        out_file_content.

% print list of files
out_list:-
    file(Pos,Name),
    write(Pos, " ", Name), nl, fail.
out_list:-!.

% out curret loaded file content
out_file_content:-
    word(Name),
    write(Name," "), fail.
out_file_content:- nl.

% load file content
read_file(FileName):-
    file_str(FileName, S),
    divide(S),
    save("word.dba").

% split one string to words
divide(S):-
    fronttoken(S, Token, Rest),
    assert(word(Token)),
    storage(A,B,C),
    write(A), readchar(_), nl,
    !,
    divide(Rest).
divide(_):-!.

% -----

% search word in loaded file
word_search:-
    nl, write("Enter word to search: "),
    readln(Sword),
    word_check(Sword),
    readln(Symb),
    file_edit(Symb, Sword).

word_check(Wch):-
    word(Wch), % proverka sushestvovania
    write("Word exist. For delete enter 'd': ").
word_check(_):- write("Word not exist. To add press 'a': ").

% redacting loaded file in database

```

```

file_edit(Symb, W):-
    Symb = "d",
    retract(word(W)).
file_edit(Symb, W):-
    Symb = "a",
    assert(word(W)).
file_edit(_, _):- write("Do nothing...").

% -----

% save edited file from database on disk
save_data:-
    currentFile(Name),

    assert(temp("")),
    my_turbo_concat(),
    temp(Result),
    write("File saved: "),
    write(Result),

    file_str(Name, Result).

% concatenation word from data base to single string
my_turbo_concat():-
    word(Text),
    temp(OldText),
    concat(OldText, Text, MidText),
    concat(MidText, " ", NewText),
    retract(temp(OldText)),
    assert(temp(NewText)),
    fail.

my_turbo_concat():- nl.

```

Пример работы программы представлен на рисунках 3-6:

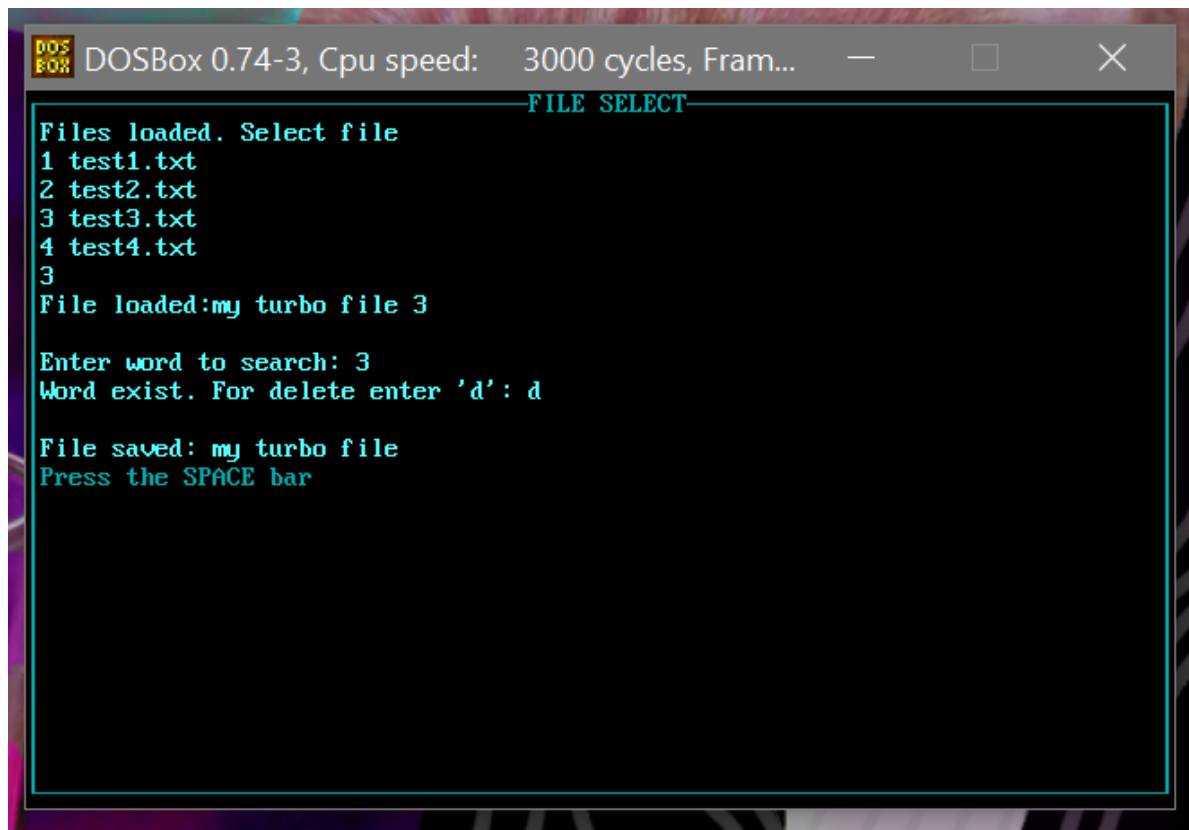


Рисунок 3 – Выбор файла номер 3, поиск в нем слова «3» и его удаление.

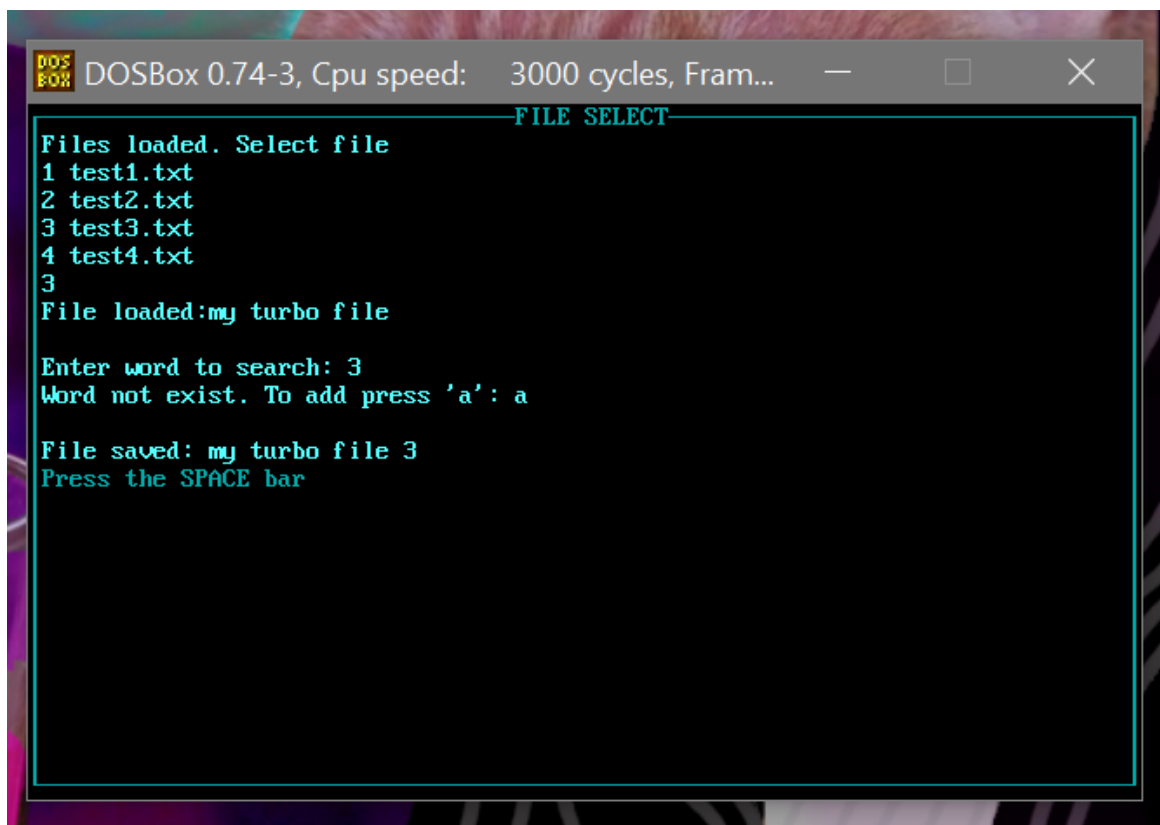


Рисунок 4 – Выбор файла номер 3, поиск в нем слова «3» и так как его в файле уже нет, добавление.

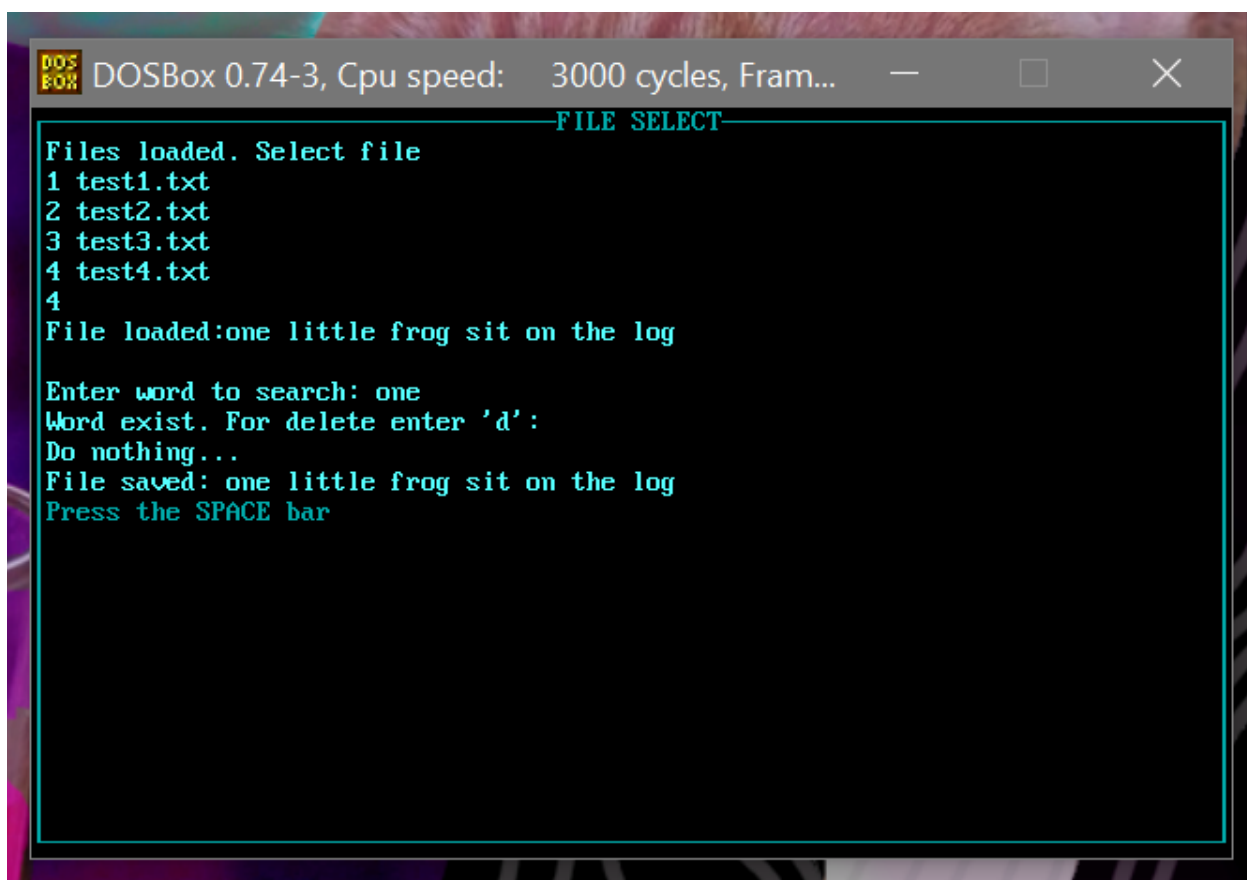


Рисунок 5 – Выбор файла номер 4, поиск в нем слова «one» и просто просмотр содержимого.

Вывод: в ходе данной лабораторной работы осуществлено ознакомление с базовыми функциями и принципами использования языка Turbo Prolog. Было осуществлено проектирование структуры и реализация программы для решения, поставленной задачи на данном языке.