Министерство науки и высшего образования Российской Федерации

Федерального государственного бюджетного образовательного учреждения высшего образования

Московский государственный технический университет имени Н.Э. Баумана

(Национальный исследовательский университет)

Московский техникум космического приборостроения

**ЛАБОРАТОРНАЯ РАБОТА №9**

По теме: «Меню работы с файлами»

Специальность: 09.02.07 Информационные системы и программирование

— квалификация «Программист»

Группа ТИП-31

Проверил Петренко Л.Б.

Разработал Симонян П.Р.

Москва 2020

СОДЕРЖАНИЕ

1 Постановка задачи3

2 Схемы алгоритма программы3

2.1 Схемы алгоритма основной программы4

2.2 Схемы алгоритма подпрограммы startSetting5

2.3 Схемы алгоритма подпрограммы filesexist6

2.4 Схемы алгоритма подпрограммы checkfile7

2.5 Схемы алгоритма подпрограммы outpfile8

2.6 Схемы алгоритма подпрограммы addfile9

2.7 Схемы алгоритма подпрограммы createfiles10

2.8 Схемы алгоритма подпрограммы readfiles11

2.9 Схемы алгоритма подпрограммы addelem12

2.10 Схемы алгоритма подпрограммы delfile16

2.11 Схемы алгоритма подпрограммы editelem17

2.12 Схемы алгоритма подпрограммы delelem18

2.13 Схемы алгоритма подпрограммы deletion19

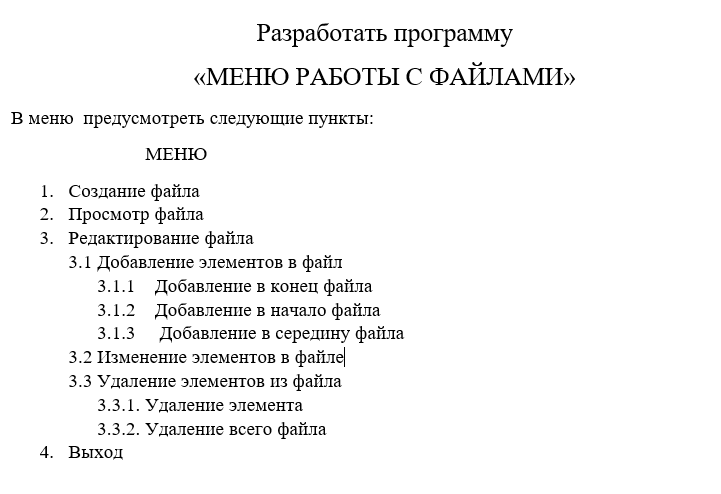
2.14 Схемы алгоритма подпрограммы editfiles20

2.15 Схемы алгоритма подпрограммы menu21

3 Листинг программы22

4 Результаты программы31

1. Постановка задачи



2. Схема алгоритма программ

2.1 Схема алгоритма основной программы



2.2 Схема алгоритма подпрограммы startSetting



2.3 Схема алгоритма подпрограммы filesexist



2.4 Схема алгоритма подпрограммы checkfile



2.5 Схема алгоритма подпрограммы outpfile



2.6 Схема алгоритма подпрограммы addfile



2.7 Схема алгоритма подпрограммы createfiles



2.8 Схема алгоритма подпрограммы readfiles



2.9 Схема алгоритма подпрограммы addelem









2.10 Схема алгоритма подпрограммы delfile



2.11 Схема алгоритма подпрограммы editelem



2.12 Схема алгоритма подпрограммы delelem



2.13 Схема алгоритма подпрограммы deletion



2.14 Схема алгоритма подпрограммы editfiles



2.15 Схема алгоритма подпрограммы menu



3. Листинг программы

**program** filemanager;

**uses** crt;

**type**

files = **array** [1..15] **of** string[10];//Хранит имена файлов

**var**

f: **file of** real;

meta: **file of** files;

menuParam: **set of** char;

filesarr: files;

path: string;

c, c1: char;

num: integer;

//Стартовая настройка программы, создание папки для хранения файлов.

**procedure** startSetting;

**begin**

assign(meta, 'PSLBData\meta.dat');

**if not** fileexists('PSLBData\meta.dat') **then**

**begin**

textcolor(red);

writeln('Данная программа создает папку под названием PSLBData для хранения созданных вами файлов.');

writeln('Для отмены создания закройте программу. Если вы хотите продолжить, нажмите любую клавишу.');

writeln('Данная процедура выполняется только один раз.');

textcolor(black);

readkey;

createdir('PSLBData');

rewrite(meta);

write(meta, filesarr);

close(meta);

**end**

**else begin**

reset(meta);

read(meta, filesarr);

close(meta);

**end**;

menuParam := ['0', '1', '2', '3'];

**end**;

//Вывод созданных файлов

**procedure** filesexist;

**var**

i: integer;

**begin**

**for** i := 1 **to** 15 **do**

**begin**

**if not** (filesarr[i] = '') **then**

writeln(i, ': ', filesarr[i]);

**end**;

**end**;

//Проверка файла на существование

**function** checkfile(name: string): boolean;

**var**

i: integer;

**begin**

checkfile := false;

**for** i := 1 **to** 15 **do**

**begin**

**if** filesarr[i] = name **then**

checkfile := true;

**end**;

**end**;

//Вывод содержимого файла

**procedure** outpfile(num: integer);

**var**

buf: real;

**begin**

writeln('Содержимое файла:');

assign(f, path + filesarr[num] + '.pdat');

reset(f);

**while not** eof(f) **do**

**begin**

read(f, buf);

write(buf, ' ');

**end**;

close(f);

**end**;

//Добавление файла в список созданных файлов

**function** addfile(name: string): boolean;

**var**

i: integer;

flag: boolean;

**begin**

i := 1;

flag := false;

**repeat**

**if** filesarr[i] = '' **then begin**

flag := true;

**end**;

i += 1;

**until** (i > 15) **or** flag;

**if** flag = false **then**

addfile := false

**else begin**

filesarr[i - 1] := name;

rewrite(meta);

write(meta, filesarr);

close(meta);

addfile := true;

**end**;

**end**;

//Создание файла

**procedure** createfiles;

**var**

name: string[10];

b: boolean;

**begin**

**repeat**

b := false;

clrscr;

textcolor(green); writeln('Создание файлов'); textcolor(black);

writeln('Введите название файла, который вы хотите создать (до 10 символов)');

writeln('Или нажмите Enter, оставив название пустым,чтобы прервать создание файла.');

readln(name);

**if not** (name = '') **then begin**

b := checkfile(name);

**if** b **then begin**

clrscr;

writeln('Файл с таким именем уже существует. Попробуйте использовать другое имя.');

readkey;

**end**;

**end**;

**until not** b;

**if not** (name = '') **then begin**

**if not** (addfile(name)) **then**

writeln('Файл не был создан, так как закончилось место (создано 15 файлов). Удалите какой-нибудь файл и попробуйте снова.')

**else begin**

assign(f, path + name + '.pdat');

rewrite(f);

close(f);

clrscr;

writeln('Файл успешно создан. Для его редактирования используйте третий пункт меню.');

**end**;

readkey;

**end**;

**end**;

//Чтение файлов

**procedure** readfiles;

**var**

buf: real;

**begin**

**repeat**

clrscr;

textcolor(green); writeln('Просмотр файлов'); textcolor(black);

writeln('Введите номер файла, который вы хотите просмотреть, или 0 для выхода');

writeln;

writeln('Созданные файлы:');

filesexist;

gotoxy(1, 3);

readln(num);

**until** (num < 16) **and** (num >= 0);

**if not** (num = 0) **then begin**

clrscr;

**if** filesarr[num] = '' **then**

writeln('Файла с таким номером не существует. Попробуйте еще раз.')

**else begin**

outpfile(num);

**end**;

readkey;

**end**;

**end**;

//Записать элементы в файл

**procedure** addelem;

**var**

buf: real;

filebuf: **file of** real;

n, i: integer;

**begin**

clrscr;

writeln('Куда добавить новые элементы?');

writeln('1: В конец файла');

writeln('2: В начало файла');

writeln('3: В середину файла');

writeln('0: Выход');

**repeat**

c1 := readkey;

**until** c1 **in** menuparam;

clrscr;

**case** c1 **of**

'1':

**begin**

reset(f);

writeln('Изменяемый файл:');

**while not** eof(f) **do**

**begin**

read(f, buf);

write(buf, ' ');

**end**;

writeln; writeln;

writeln('Вводите новые элементы через Enter. Чтобы завершить ввод, нажмите Ctrl + Z или F6, затем еще раз Enter');

**while not** eof **do**

**begin**

readln(buf);

write(f, buf);

**end**;

flush(input);

close(f);

clrscr;

textcolor(green); writeln('Файл успешно изменен!'); textcolor(black);

readkey;

**end**;

'2':

**begin**

reset(f);

assign(filebuf, path + 'temp.dat');

rewrite(filebuf);

writeln('Изменяемый файл:');

**while not** eof(f) **do**

**begin**

read(f, buf);

write(filebuf, buf);

write(buf, ' ');

**end**;

writeln; writeln;

writeln('Вводите новые элементы через Enter. Чтобы завершить ввод, нажмите Ctrl + Z или F6, затем еще раз Enter');

rewrite(f);

**while not** eof **do**

**begin**

readln(buf);

write(f, buf);

**end**;

reset(filebuf);

**while not** eof(filebuf) **do**

**begin**

read(filebuf, buf);

write(f, buf);

**end**;

flush(input);

close(f);

close(filebuf);

clrscr;

textcolor(green); writeln('Файл успешно изменен!'); textcolor(black);

readkey

**end**;

'3':

**begin**

i := 0;

n := 0;

reset(f);

assign(filebuf, path + 'temp.dat');

rewrite(filebuf);

writeln('Изменяемый файл:');

**while not** eof(f) **do**

**begin**

i += 1;

read(f, buf);

write(buf, ' ');

write(filebuf, buf);

**end**;

reset(filebuf);

rewrite(f);

n := round(i / 2);

i := 0;

writeln; writeln;

writeln('Вводите новые элементы через Enter. Чтобы завершить ввод, нажмите Ctrl + Z или F6, затем еще раз Enter');

**if not** (n = 0) **then begin**

**repeat**

i += 1;

read(filebuf, buf);

write(f, buf);

**if** i = n **then begin**

**while not** eof **do**

**begin**

readln(buf);

write(f, buf);

**end**;

**end**;

**until** eof(filebuf);

**end**

**else begin**

**while not** eof **do**

**begin**

readln(buf);

write(f, buf);

**end**;

**end**;

flush(input);

close(f);

close(filebuf);

clrscr;

textcolor(green); writeln('Файл успешно изменен!'); textcolor(black);

readkey;

**end**;

'0': clrscr;

**end**;

readln;

**end**;

//Удаление файла

**procedure** delfile;

**begin**

clrscr;

writeln('Вы действительно хотите удалить файл под названием "', filesarr[num], '"?');

textcolor(red); writeln('1: Да');

textcolor(green); writeln('2: Нет'); textcolor(black);

writeln;

outpfile(num);

**repeat**

c1 := readkey;

**until** c1 **in** ['1', '2'];

clrscr;

**case** c1 **of**

'1':

**begin**

deletefile(path + filesarr[num] + '.pdat');

filesarr[num] := '';

rewrite(meta);

write(meta, filesarr);

close(meta);

writeln('Файл успешно удален');

**end**;

'2': writeln('Удаление файла отменено');

**end**;

readkey;

**end**;

//Изменение элементов

**procedure** editelem;

**var**

filebuf: **file of** real;

buf: real;

**begin**

clrscr;

writeln('Данная функция отбрасывает дробную часть чисел в файле.');

writeln('К примеру: 2.51 => 2');

writeln('Нажмите 0, если хотите отменить операцию. Для продолжения нажмите другую клавишу.');

writeln; outpfile(num);

c1 := readkey;

**if not** (c1 = '0') **then begin**

clrscr;

outpfile(num);

writeln;

assign(filebuf, path + 'temp.dat');

rewrite(filebuf);

reset(f);

**while not** eof(f) **do**

**begin**

read(f, buf);

write(filebuf, trunc(buf));

**end**;

reset(filebuf);

rewrite(f);

writeln('Обработанный файл:');

**while not** eof(filebuf) **do**

**begin**

read(filebuf, buf);

write(f, buf);

write(buf, ' ');

**end**;

close(filebuf);

close(f);

writeln;

writeln('Все найденные дробные числа изменены. Нажмите любую клавишу, чтобы продолжить.');

readkey;

**end**;

**end**;

//Удаление элементов

**procedure** delelem;

**var**

filebuf: **file of** real;

buf: real;

**begin**

clrscr;

writeln('Данная функция удаляет нулевые значения файла.');

writeln('Нажмите 0, если хотите отменить операцию. Для продолжения нажмите другую клавишу.');

writeln; outpfile(num);

c1 := readkey;

**if not** (c1 = '0') **then begin**

clrscr;

outpfile(num);

writeln;

assign(filebuf, path + 'temp.dat');

rewrite(filebuf);

reset(f);

**while not** eof(f) **do**

**begin**

read(f, buf);

**if not** (buf = 0) **then**

write(filebuf, buf);

**end**;

reset(filebuf);

rewrite(f);

writeln('Обработанный файл:');

**while not** eof(filebuf) **do**

**begin**

read(filebuf, buf);

write(buf, ' ');

write(f, buf);

**end**;

close(filebuf);

close(f);

writeln;

writeln('Все найденные нулевые элементы удалены. Нажмите любую клавишу, чтобы продолжить.');

readkey;

**end**;

**end**;

//Меню удаления

**procedure** deletion;

**begin**

clrscr;

writeln('Выберите режим удаления');

writeln('1: Удаление нулевых элементов файла');

writeln('2: Удаление файла целиком');

writeln('0: Выход');

menuparam -= ['3'];

**repeat**

c1 := readkey;

**until** c1 **in** menuparam;

menuparam += ['3'];

writeln(23);

**case** c1 **of**

'1': delelem;

'2': delfile;

'0': clrscr;

**end**;

**end**;

//Меню редактирования файлов

**procedure** editfiles;

**begin**

**repeat**

clrscr;

textcolor(green); writeln('Просмотр файлов'); textcolor(black);

writeln('Введите номер файла, который вы хотите отредактировать, или 0 для выхода');

writeln;

writeln('Созданные файлы:');

filesexist;

gotoxy(1, 3);

readln(num);

**until** (num < 16) **and** (num >= 0);

**if not** (num = 0) **then begin**

clrscr;

**if** filesarr[num] = '' **then begin**

writeln('Файла с таким номером не существует. Попробуйте еще раз.');

readkey;

**end**

**else begin**

assign(f, path + filesarr[num] + '.pdat');

writeln('Выберите режим редактирования');

writeln;

writeln('1: Добавление элементов в файл');

writeln('2: Изменение элементов файла');

writeln('3: Удаление элементов файла');

writeln('0: Выход');

**repeat**

c1 := readkey;

**until** c1 **in** menuparam;

**case** c1 **of**

'1': addelem;

'2': editelem;

'3': deletion;

'0': clrscr;

**end**;

**end**;

**end**;

**end**;

//Главное меню

**procedure** menu;

**begin**

clrscr;

writeln('Меню управления файлами');

writeln;

writeln('1. Создание файлов');

writeln('2. Просмотр файлов');

writeln('3. Редактирование файлов');

writeln('0. Выход');

gotoxy(1, 10);

textcolor(red);

writeln('Все операции с файлами, созданными с помощью этой программы, должны осуществляться только через это меню!');

writeln('В случае ошибок, вызванных непредусмотренным доступом к данным (например, ручное удаление файлов в проводнике Windows)');

writeln('необходимо удалить папку PSLBData в месте установки программы, а затем перезапустить ее. Все данные сотрутся.');

textcolor(black);

**repeat**

c := readkey;

**until** c **in** menuparam;

**case** c **of**

'1': createfiles;

'2': readfiles;

'3': editfiles;

'0': clrscr;

**end**;

**end**;

**begin**

textcolor(black); textbackground(lightgray); clrscr;

startSetting;

path := 'PSLBData\';

**repeat**

menu;

**until** c = '0';

**end**.

4 Результаты программы

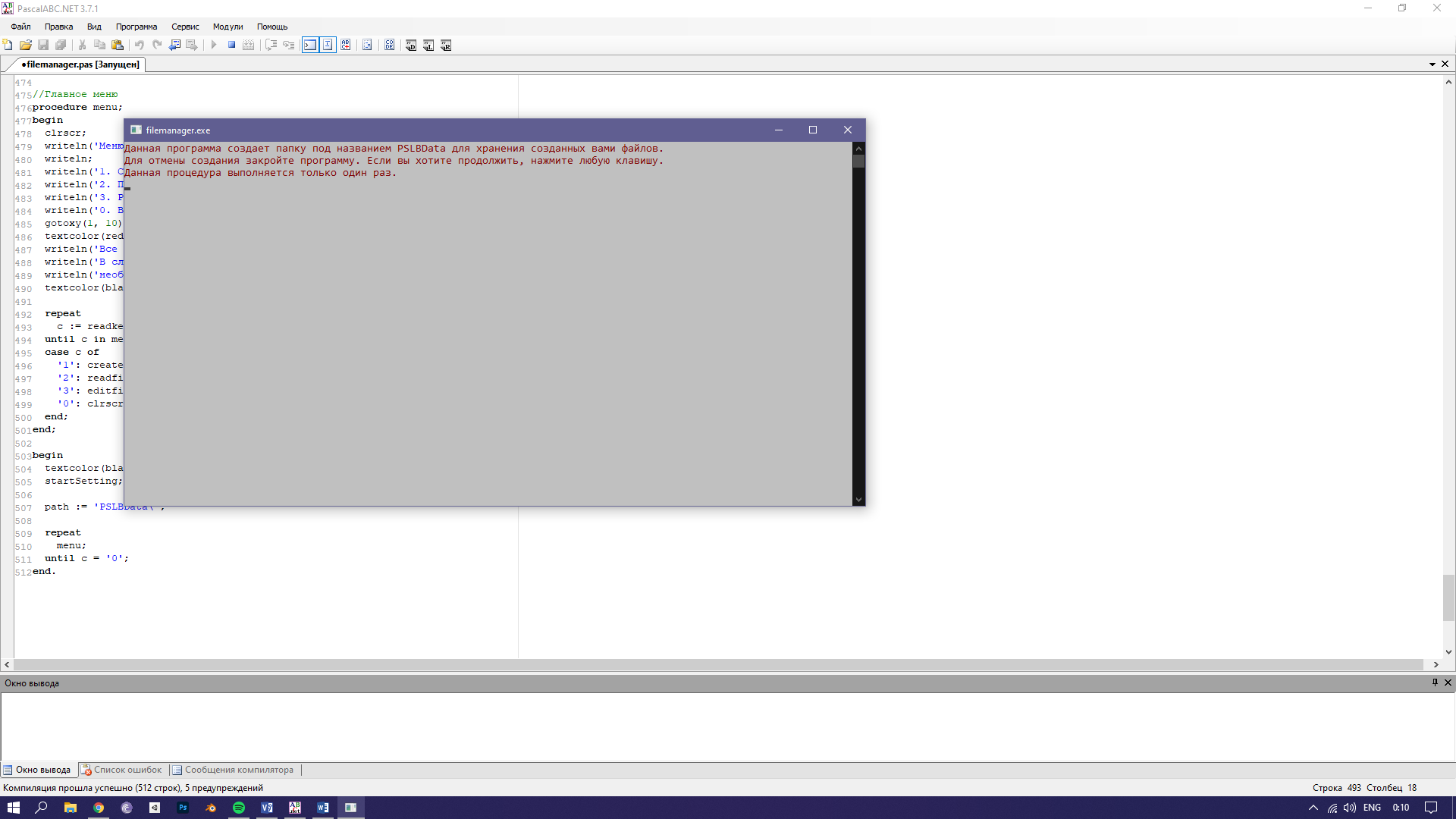


Рис. 1

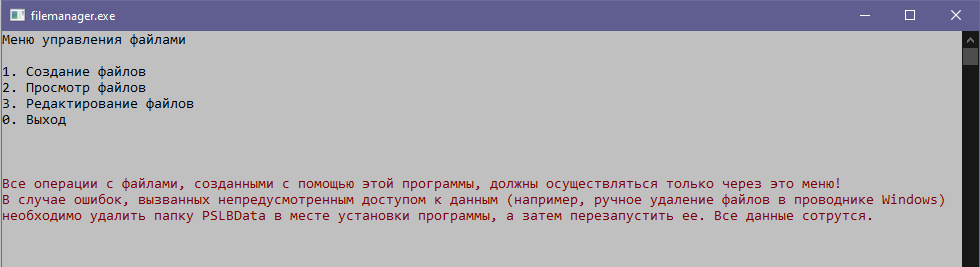


Рис. 2

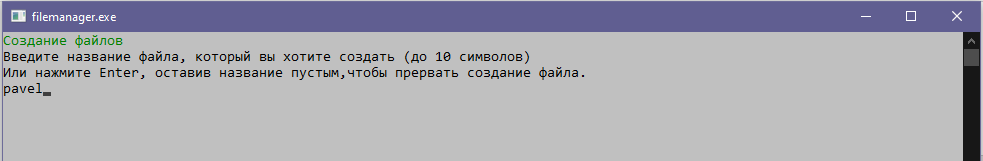


Рис. 3

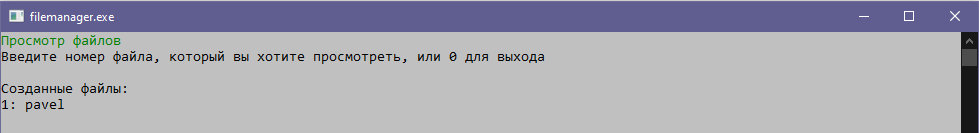


Рис. 4

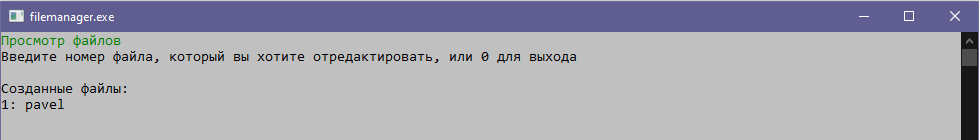


Рис. 5



Рис. 6



Рис. 7

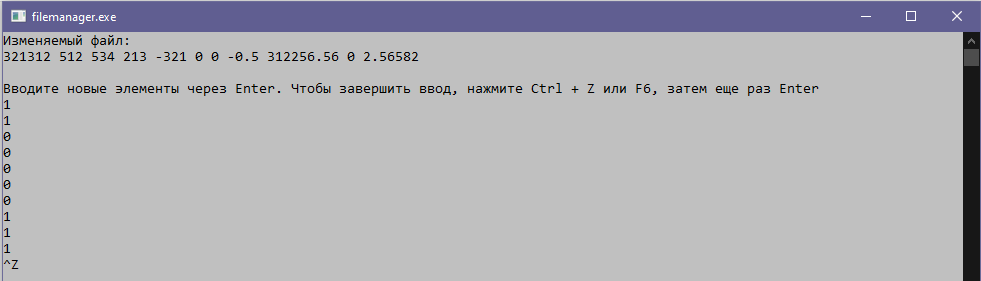


Рис. 8 (Добавление в середину)

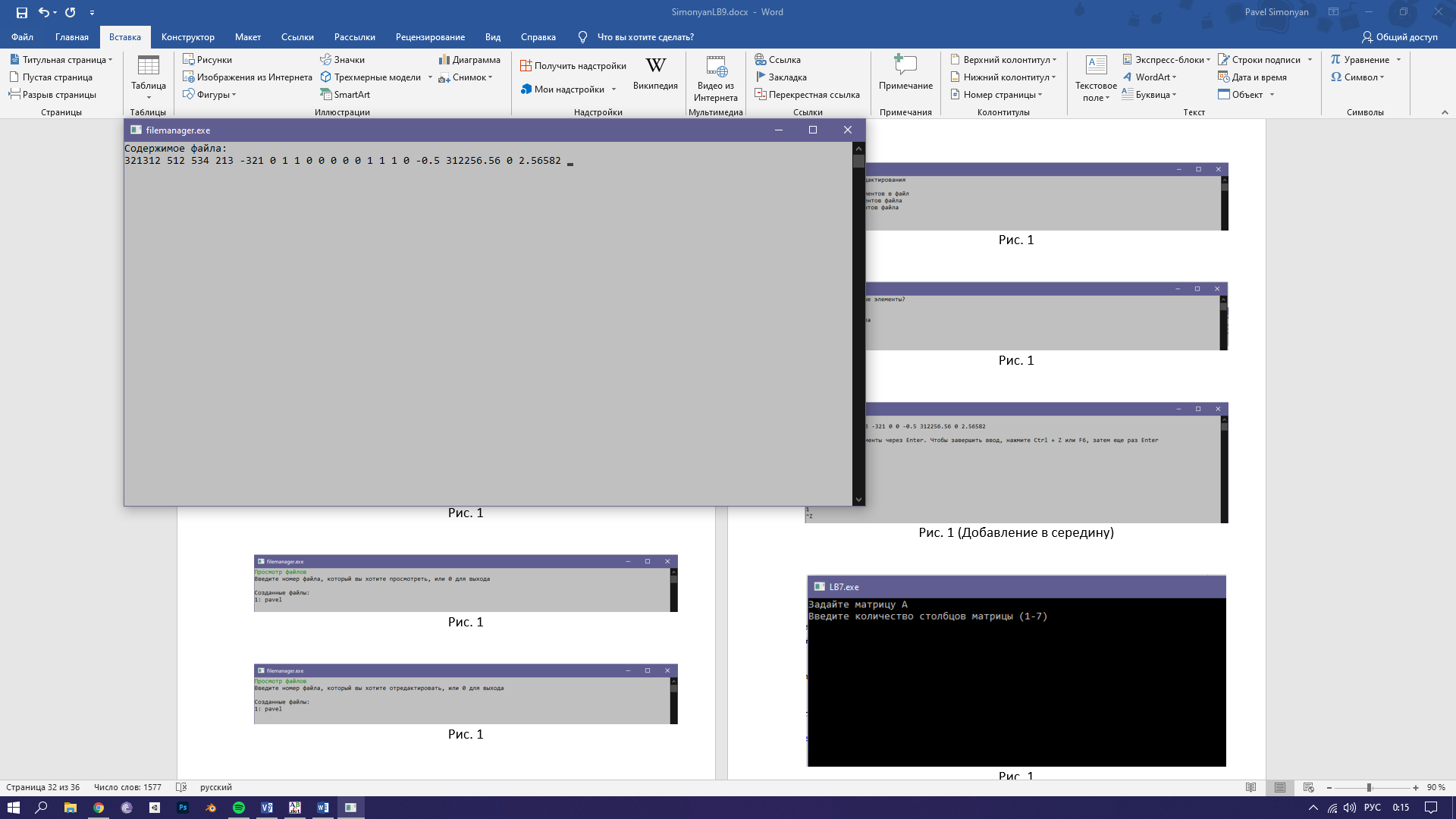


Рис. 9 (Просмотр файла)

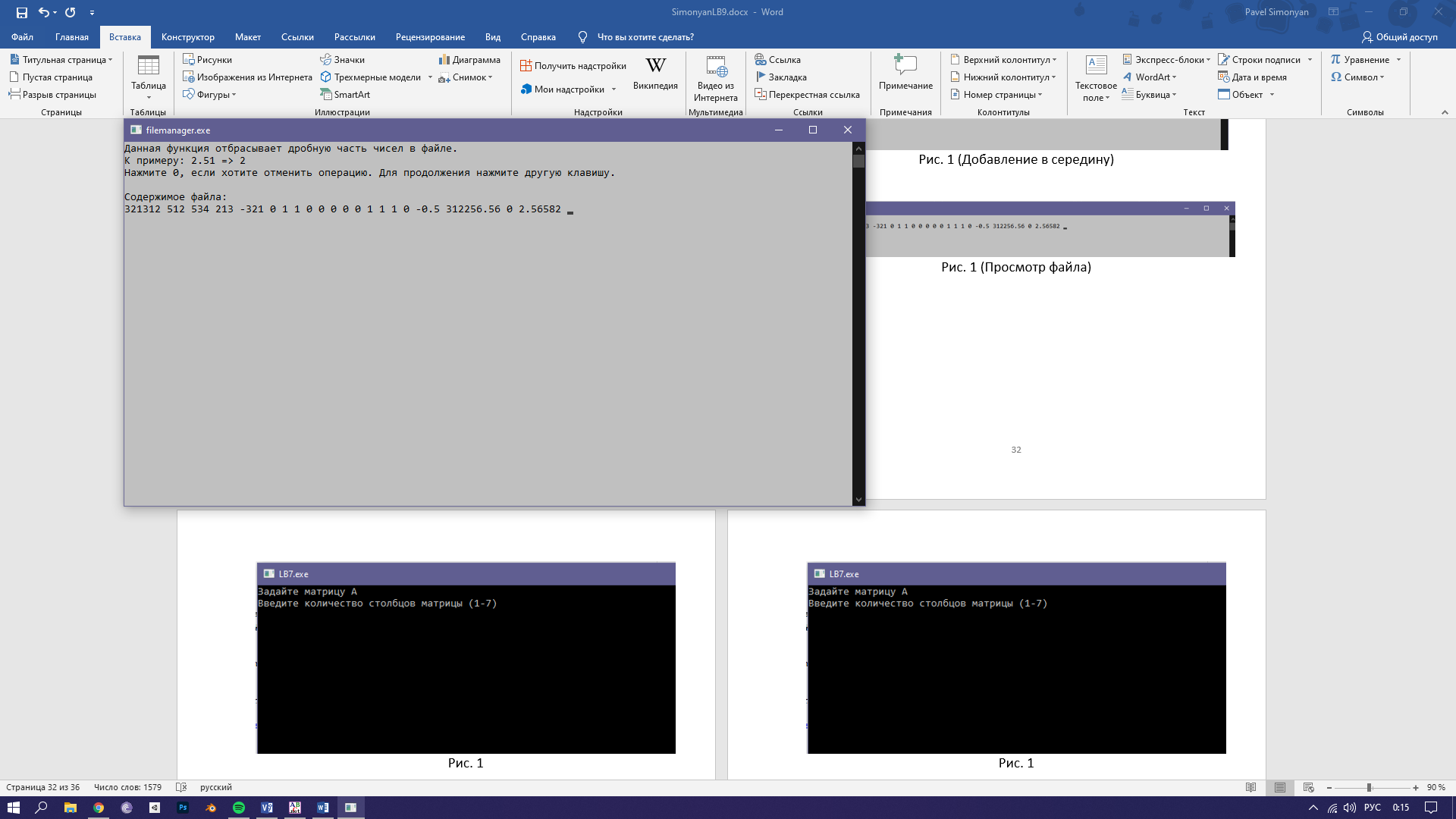


Рис. 10 (Редактирование элементов)

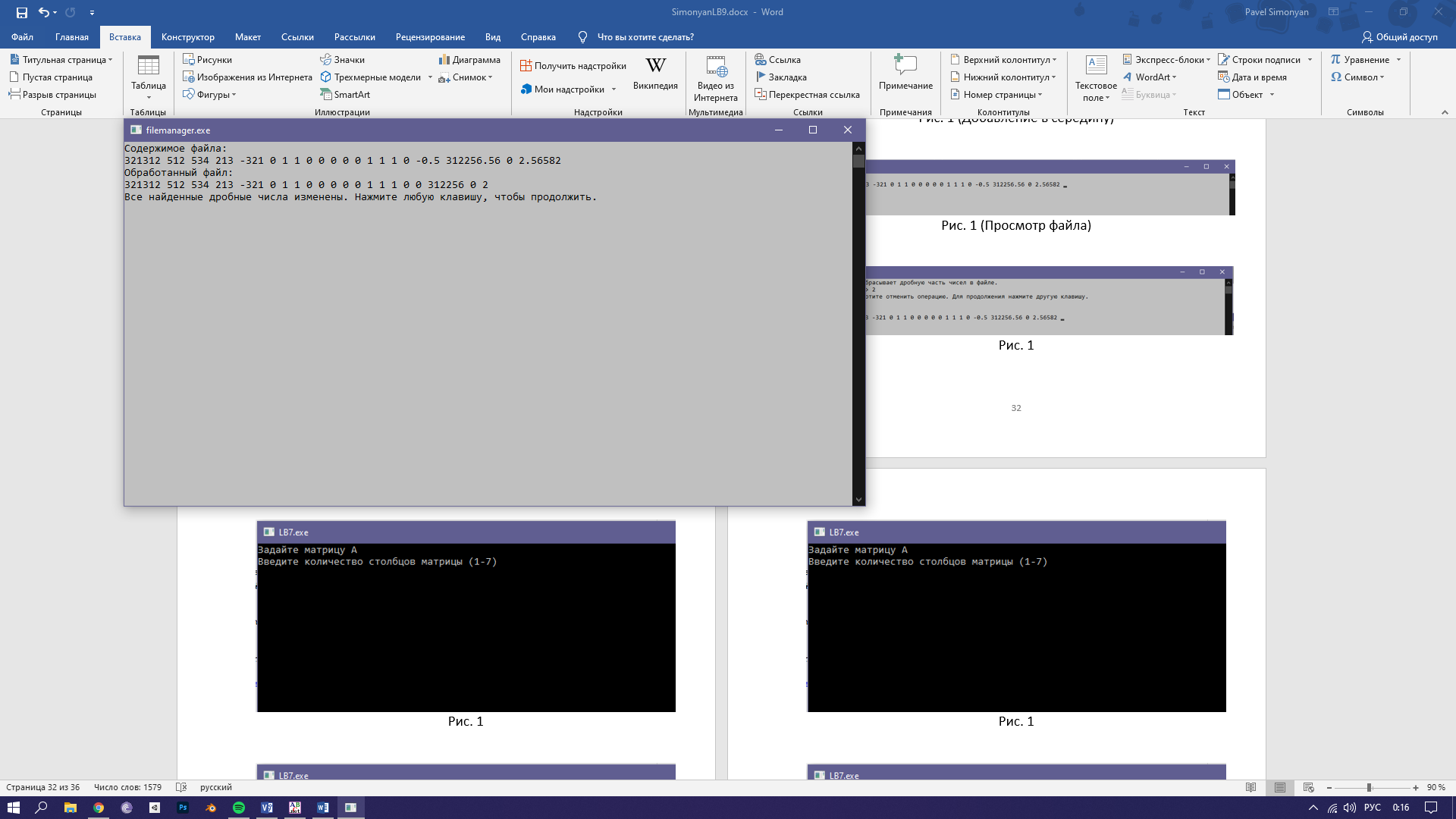


Рис. 11 (Редактирование элементов)



Рис. 12 (Удаление)

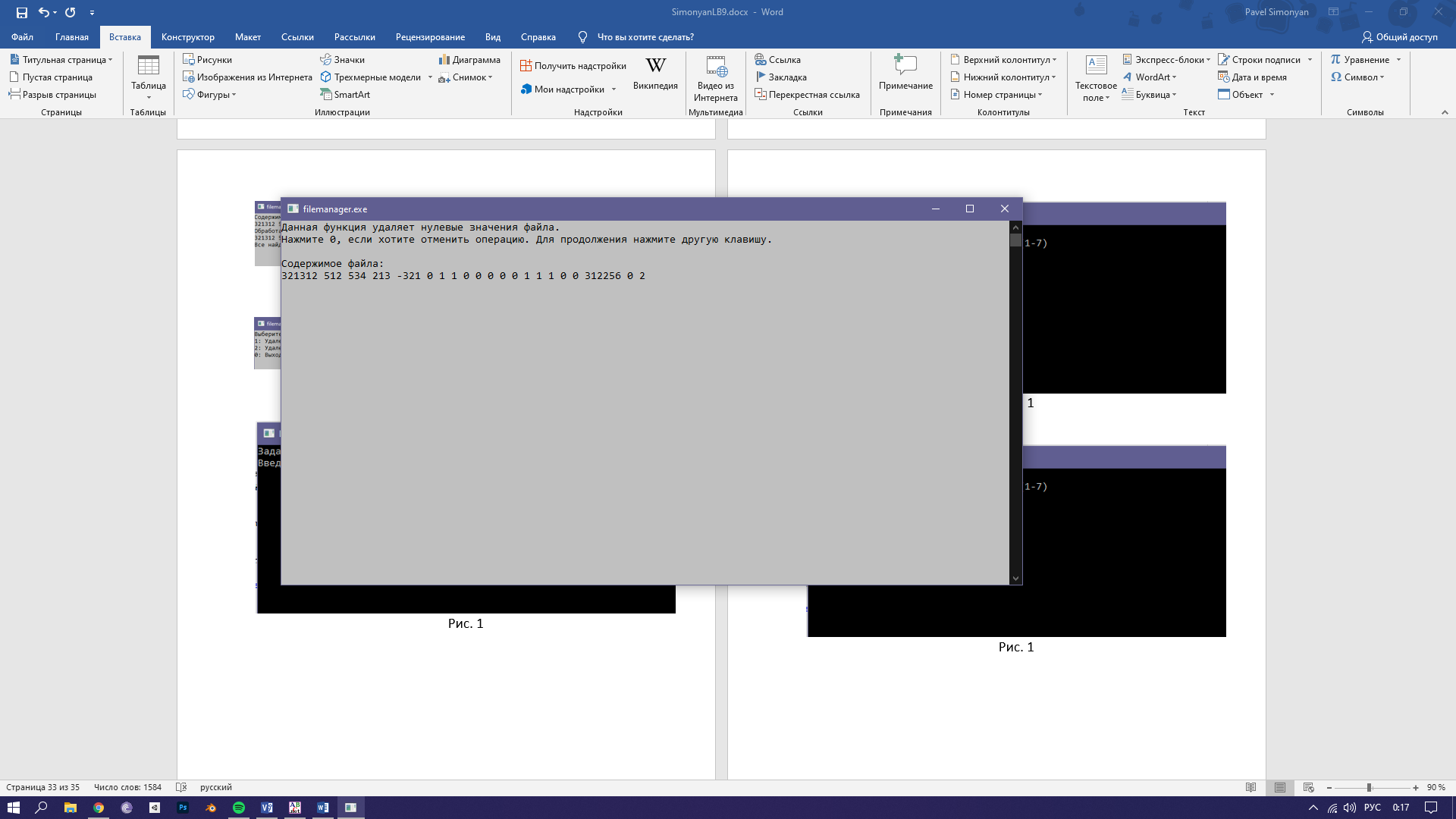


Рис. 13 (Удаление элементов)

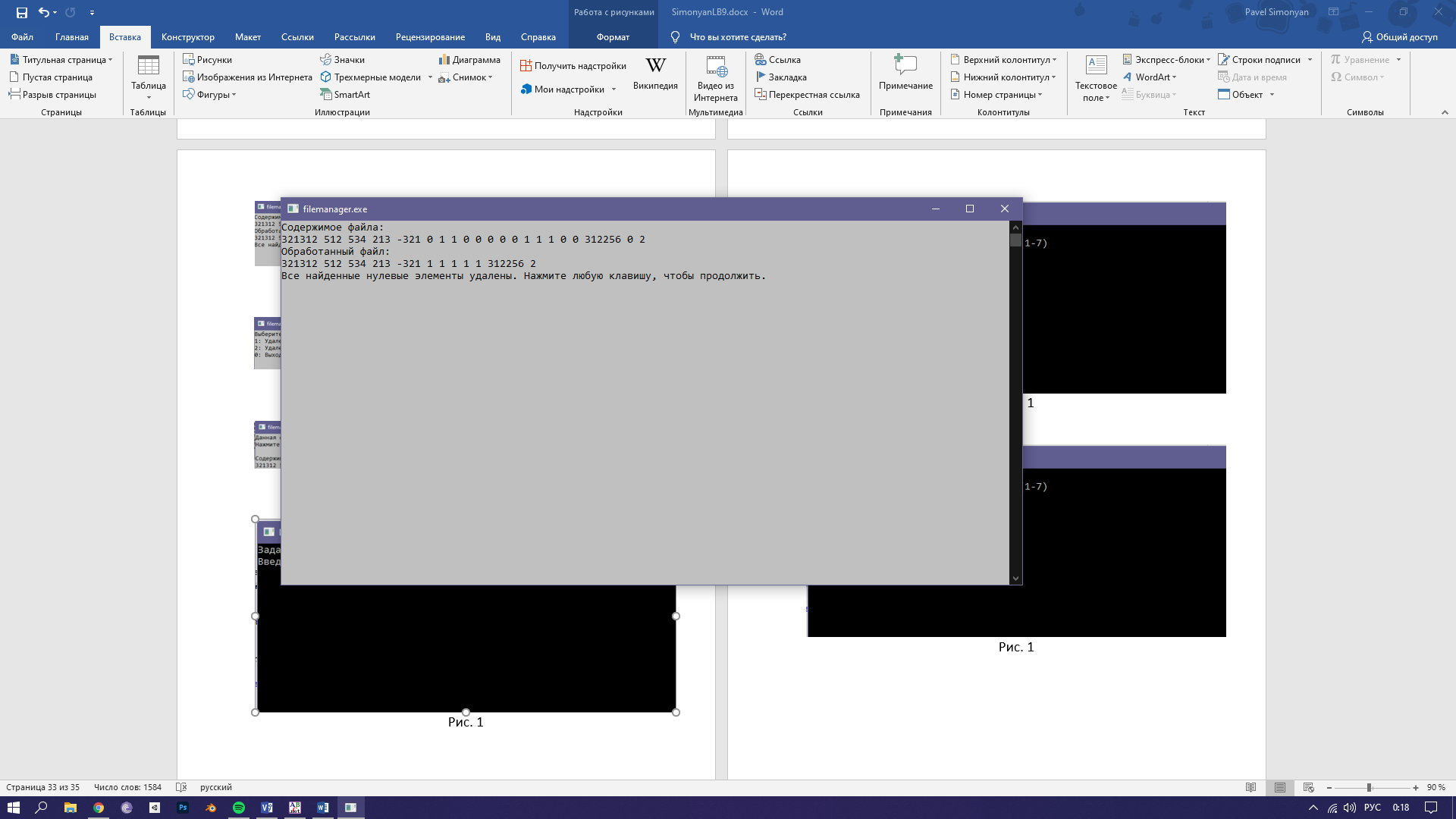


Рис. 14 (Удаление элементов)

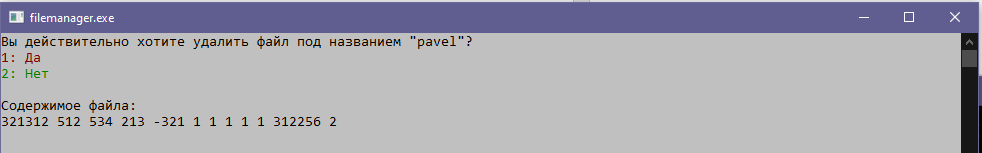


Рис. 15 (Удаление файла)

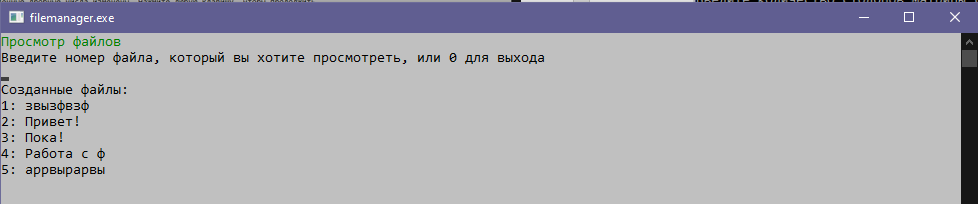


Рис. 16

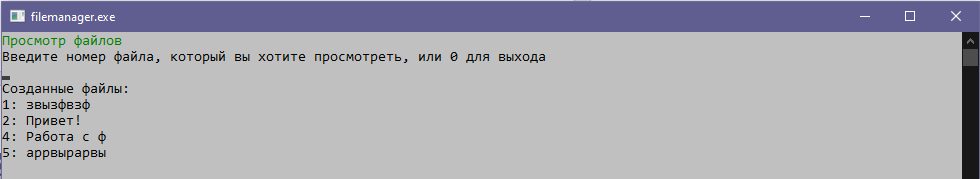


Рис. 17 (После удаления 3 файла)