Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №5

по курсу «Программирование»

на тему «Процедуры»

Вариант 7

Выполнили:

студенты группы 24ВВВ1

Тусков А.А.

Цыбузин Д.В.

Кустов И.Д.

Приняли:

к.т.н., доцент Токарев А.Н.

к.т.н., доцент Генералова А.А.

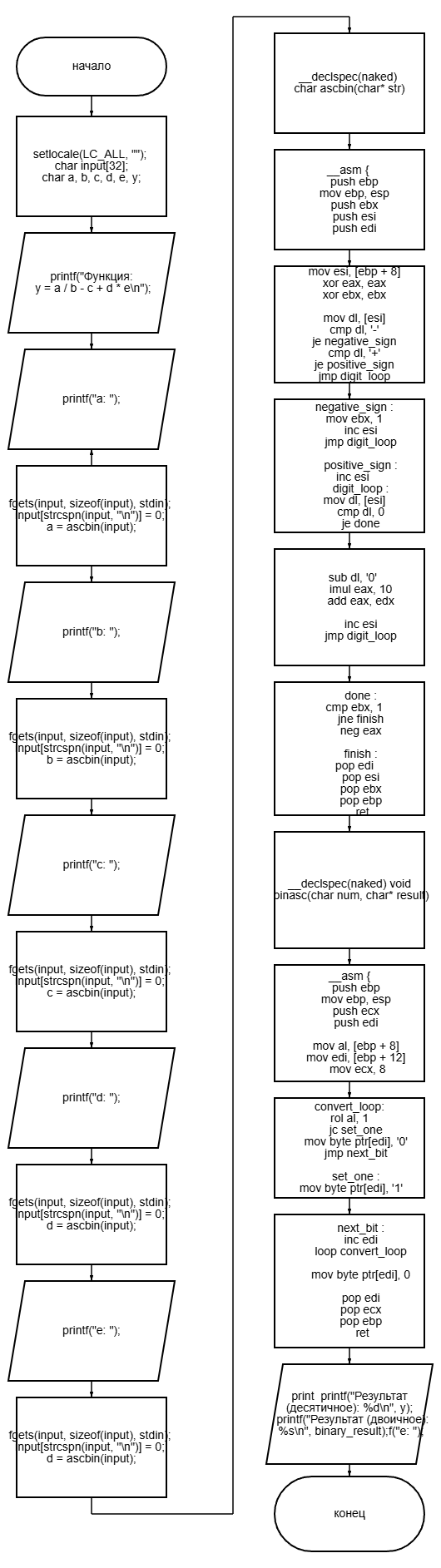
Пенза 2024

**Цель работы:**получение навыков разработки структурной организации ассемблерных программ и ее реализации на основе аппарата процедур, изучение способов организации связи по данным между процедурами.

**Лабораторное задание:**

Разработать программу вычисления функции y = a / b - c + d \* e\n *.* Значения аргументов *a, b, c, d, e* должны вводиться с клавиатуры ПЭВМ. Значение функции *y* следует вывести на экран дисплея.

**Описание метода решения:**

****

**Листинг:**

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <limits.h>

#include <locale.h>

char binary\_result[9]; // 8 бит + нулевой терминатор

\_\_declspec(naked) char ascbin(char\* str) {

\_\_asm {

push ebp

mov ebp, esp

push ebx

push esi

push edi

mov esi, [ebp + 8]

xor eax, eax

xor ebx, ebx

mov dl, [esi]

cmp dl, '-'

je negative\_sign

cmp dl, '+'

je positive\_sign

jmp digit\_loop

negative\_sign :

mov ebx, 1

inc esi

jmp digit\_loop

positive\_sign :

inc esi

digit\_loop :

mov dl, [esi]

cmp dl, 0

je done

sub dl, '0'

imul eax, 10

add eax, edx

inc esi

jmp digit\_loop

done :

cmp ebx, 1

jne finish

neg eax

finish :

pop edi

pop esi

pop ebx

pop ebp

ret

}

}

\_\_declspec(naked) void binasc(char num, char\* result) {

\_\_asm {

push ebp

mov ebp, esp

push ecx

push edi

mov al, [ebp + 8]

mov edi, [ebp + 12]

mov ecx, 8

convert\_loop:

rol al, 1

jc set\_one

mov byte ptr[edi], '0'

jmp next\_bit

set\_one :

mov byte ptr[edi], '1'

next\_bit :

inc edi

loop convert\_loop

mov byte ptr[edi], 0

pop edi

pop ecx

pop ebp

ret

}

}

\_\_declspec(naked) char func(char a, char b, char c, char d, char e) {

\_\_asm {

push ebp

mov ebp, esp

push ebx

mov al, [ebp + 8] // a

mov bl, [ebp + 12] // b

// Вычисляем a / b

cbw

idiv bl

// Вычитаем c

sub al, [ebp + 16] // c

// Вычисляем d \* e

mov dl, [ebp + 20] // d

mov bl, [ebp + 24] // e

imul dl, bl

// Добавляем d\*e к предыдущему результату

add al, dl

pop ebx

pop ebp

ret

}

}

int main() {

setlocale(LC\_ALL, "");

char input[32];

char a, b, c, d, e, y;

printf("Функция: y = a / b - c + d \* e\n");

printf("a: ");

fgets(input, sizeof(input), stdin);

input[strcspn(input, "\n")] = 0;

a = ascbin(input);

printf("b: ");

fgets(input, sizeof(input), stdin);

input[strcspn(input, "\n")] = 0;

b = ascbin(input);

printf("c: ");

fgets(input, sizeof(input), stdin);

input[strcspn(input, "\n")] = 0;

c = ascbin(input);

printf("d: ");

fgets(input, sizeof(input), stdin);

input[strcspn(input, "\n")] = 0;

d = ascbin(input);

printf("e: ");

fgets(input, sizeof(input), stdin);

input[strcspn(input, "\n")] = 0;

e = ascbin(input);

y = func(a, b, c, d, e);

binasc(y, binary\_result);

printf("Результат (десятичное): %d\n", y);

printf("Результат (двоичное): %s\n", binary\_result);

return 0;

}

**Результат работы программы:**

**Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.**

**Протокол трассировки:**

На рисунке 1 а преобразуется в число

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1

На рисунке 2 полученная строка, с введённым значением b, преобразована в число

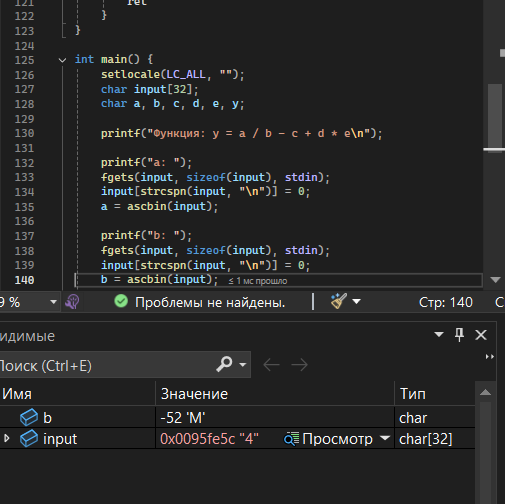


Рисунок 2

На рисунке 3 считывается введенная строка, чтобы потом преобразовать в число.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 3