Федеральное государственное бюджетное образовательное учреждение высшего образования «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича»

Кафедра «Программная инженерия и вычислительная техника»

«Машино-зависимые языки программирования»

Отчет

по лабораторной работе №1

«Вычисление целочисленных арифметических выражений»

|  |  |
| --- | --- |
|  | Выполнила |
|  | студентка группы ИКПИ-25 |
|  | А. Ю. Яскунова |
|  |  |
|  | Проверил |
|  | Ст. преподаватель |
|  | И.Л. Савельев |
|  |  |

2023 г.

**1. Задание**

Вычислить заданное целочисленное выражение для исходных данных в знаковых и беззнаковых форматах длиной 8 и 16 бит: signed char, unsigned int , используя арифметические операции ADD, ADC, INC, SUB, SBB , DEC, NEG, MUL, IMUL, DIV, IDIV , CBW, CWD. Исходные значения переменных вводятся пользователем с клавиатуры. Они должны быть максимально приближены к максимально-возможным для тех типов данных, с которыми решается задача. При вводе данных рекомендуется вывести диапазон возможных значений. Размер и тип числителя, знаменателя и результата зависит от заданного выражения.

**Вариант №26**

(-3\*a-b+53)/(c-a/2+1)

**2. Текст программы**

**2.1 int8**

**2.1.1 Модуль int8.c**

#include <stdio.h>

#include <stdint.h>

#include <stdlib.h>

int8\_t a, b, c;

int numerator\_c, denominator\_c, result\_c;

int numerator\_asm, remainder\_asm, denominator\_asm, result\_asm;

extern void asmfunc();

int main (void)

{

printf("(-3\*a-b+53)/(c-a/2+1)");

printf("\nEnter a from -128 to 127 = ");

scanf("%hhd", &a);

printf("Enter b from -128 to 127 = ");

scanf("%hhd", &b);

printf("Enter c from -128 to 127 = ");

scanf("%hhd", &c);

numerator\_c = -1 \* 3 \* a - b + 53;

denominator\_c = c - a/2 + 1;

result\_c = numerator\_c / denominator\_c;

int remainder = numerator\_c % denominator\_c;

printf("\nРезультат на Си: целое = %d, остаток = %d, числитель = %d, знаменатель = %d", result\_c, remainder, numerator\_c, denominator\_c);

asmfunc();

printf("\nРезультат на ассемблере: целое = %d, остаток = %d, числитель = %d, знаменатель = %d", result\_asm, remainder\_asm, numerator\_asm, denominator\_asm);

printf("\n\n");

return 0;

}

**2.1.2 Модуль asm\_int8.asm**

section .data

extern a

extern b

extern c

extern numerator\_asm

extern denominator\_asm

extern result\_asm

extern remainder\_asm

; y = (-3\*a-b+53)/(c-a/2+1)

section .text

global asmfunc

asmfunc:

movsx eax, byte [a]

imul eax, -3 ; Умножение EAX на -3

movsx ebx, byte [b] ; Вычитание b из EAX

sub eax, ebx

add eax, 53 ; Добавление 53 к EAX

mov [numerator\_asm], eax

movsx ebx, byte [a] ; Загрузка значения a в AX

sar ebx, 1 ; Половинное деление значения a, т.е. побитовый сдвиг

movsx ecx, byte [c] ; Вычитание a/2 из с

sub ecx, ebx

inc ecx

mov [denominator\_asm], ecx

mov eax, dword [numerator\_asm]

cdq

idiv dword [denominator\_asm]

mov dword [result\_asm], eax

mov dword [remainder\_asm], edx

ret

**2.2 uint16**

**2.2.1 Модуль uint16.c**

#include <stdio.h>

#include <stdint.h>

#include <stdlib.h>

unsigned int a, b, c;

int numerator\_c, denominator\_c, result\_c;

int numerator\_asm, remainder\_asm, denominator\_asm, result\_asm;

extern void asmfunc();

int main (void)

{

printf("(-3\*a-b+53)/(c-a/2+1)");

printf("\nEnter a from 0 to 65535 = ");

scanf("%u", &a);

printf("Enter b from 0 to 65535 = ");

scanf("%u", &b);

printf("Enter c from 0 to 65535 = ");

scanf("%u", &c);

numerator\_c = -1 \* 3 \* a - b + 53;

denominator\_c = c - a/2 + 1;

result\_c = numerator\_c / denominator\_c;

int remainder = numerator\_c % denominator\_c;

printf("\nРезультат на Си: целое = %d, остаток = %d, числитель = %d, знаменатель = %d", result\_c, remainder, numerator\_c, denominator\_c);

asmfunc();

printf("\nРезультат на ассемблере: целое = %d, остаток = %d, числитель = %d, знаменатель = %d", result\_asm, remainder\_asm, numerator\_asm, denominator\_asm);

printf("\n\n");

return 0;

}

**2.2.2 Модуль asm\_uint16.asm**

section .data

extern a

extern b

extern c

extern numerator\_asm

extern denominator\_asm

extern result\_asm

extern remainder\_asm

; y = (-3\*a-b+53)/(c-a/2+1)

section .text

global asmfunc

asmfunc:

mov ax, [a] ; Загрузка значения a в AX

shr ax, 1 ; Половинное деление значения a, т.е. побитовый сдвиг

mov bx, [c]

sub bx, ax ; Вычитание a/2 из с

inc bx

mov [denominator\_asm], bx

mov eax, [a]

imul eax, -3 ; Умножение EAX на -3

sub eax, [b] ; Вычитание b из EAX

add eax, 53 ; Добавление 53 к EAX

mov [numerator\_asm], eax

mov eax, dword [numerator\_asm]

cdq

idiv dword [denominator\_asm]

mov dword [result\_asm], eax

mov dword [remainder\_asm], edx

ret

**2.3. Модуль makefile**

all:

gcc -c -g -ggdb -o uint16.o uint16.c

nasm -f elf64 -F dwarf -g asm\_uint16.asm

gcc -m64 -o uint16 uint16.o asm\_uint16.o -g -ggdb -fno-pie -no-pie

gcc -c -g -ggdb -o int8.o int8.c

nasm -f elf64 -F dwarf -g asm\_int8.asm

gcc -m64 -o int8 int8.o asm\_int8.o -g -ggdb -fno-pie -no-pie

**3. Сборка проекта**

make

**4. Выполнение программы**

**4.1. Запуск программы**

./int8

./uint16

**4.2. Входные данные**

**Для int8:**

a = 126

b= 0

c = -120

**Для uint16:**

a = 62000

b= 35234

c = 45623

**4.3. Ожидаемый результат выполнения**

**Для int8:**

целое = 1

числитель = -325

знаменатель = -182

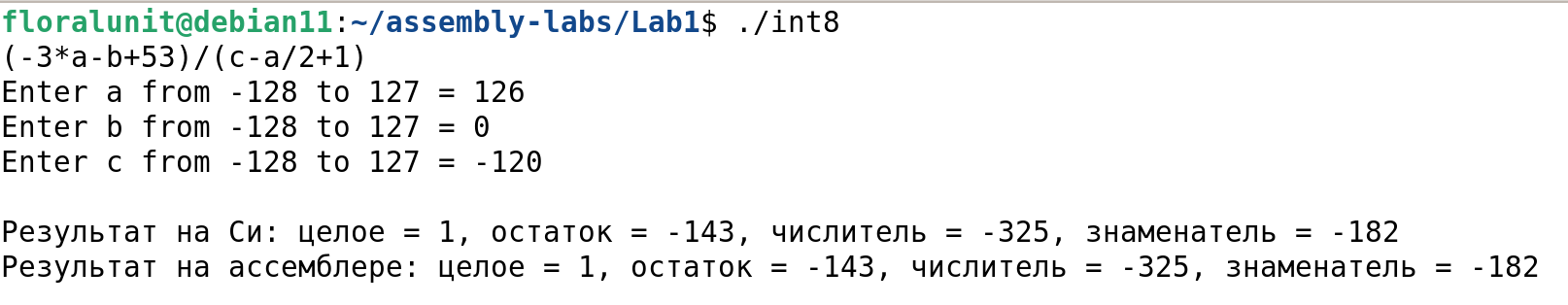
**Для uint16:**

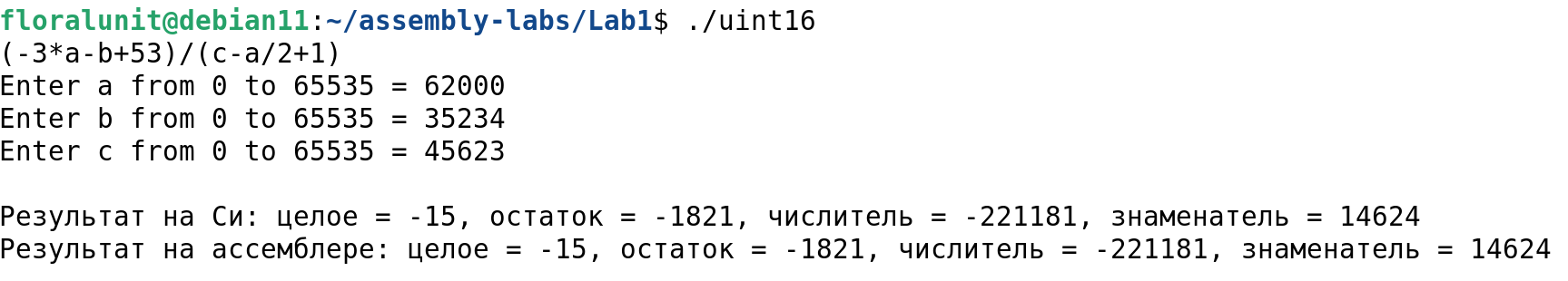
целое = -15

числитель = -221181

знаменатель = 14624

**4.4. Результат выполнения**





**5. Вывод**

Результат выполнения программы соответствует ожидаемому результату. Работа выполнена в полном объеме.