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 №5

«Сопроцессор»

|  |  |
| --- | --- |
|  | Выполнил |
|  | студент группы ИКПИ-14 |
|  | А.В.Крылов |
|  |  |
|  | Проверил |
|  | Ст. преподаватель |
|  | А.О.Овчинников |
|  |  |

2022 г.

1. Задание

Вычислить заданное вещественное выражение для исходных данных в форматах Float(SINGLE для переменных a и b) и Int(INTEGER - все остальные переменные), используя арифметические операции сопроцессора

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 №11

(2 \* d — 96 / a) / (34 / b — a + 1)

2. Текст программы

2.1. Модуль main.c

#include <stdio.h>

float num, den, res;

float a, b;

int d;

extern void afunc();

int main() {

puts(

"Zadacha №11, Krylov Artem Vyacheslavovich, IKPI-14\n"

"(2 \* d - 96 / a) / (34 / b - a + 1)\n"

"a, b: [1.2E-38; 3.4E+38]\n"

"d: [-2,147,483,648; 2,147,483,647]");

puts("Vvedite a, b i d:");

scanf("%f%f%d", &a, &b, &d);

puts("S pomoshyu ASM:");

afunc();

printf("Num = %.1f, den = %.1f, res = %.1f\n", num, den, res);

puts("S pomoshyu C:");

num = 2 \* d - 96 / a;

den = 34 / b - a + 1;

res = num / den;

printf("Num = %.1f, den = %.1f, res = %.1f\n", num, den, res);

return 0;

}

2.2. Модуль afunc.asm

extern num, den, res

extern a, b, d

global afunc

section .text

afunc:

mov eax, [d]

add eax, eax ; 2 \* d

cvtsi2ss xmm0, eax

mov eax, 96

cvtsi2ss xmm1, eax

movss xmm2, [a]

divss xmm1, xmm2 ; 96 / a

subss xmm0, xmm1 ; 2 \* d - 96 / a

movss [num], xmm0

mov eax, 34

cvtsi2ss xmm0, eax

movss xmm1, [b]

divss xmm0, xmm1 ; 34 / b

movss xmm1, [a]

subss xmm0, xmm1 ; 34 / b - a

mov eax, 1

cvtsi2ss xmm1, eax

addss xmm0, xmm1 ; 34 / b - a + 1

movss [den], xmm0

movss xmm1, [num]

divss xmm1, xmm0

movss [res], xmm1

ret

2.3. Модуль Makefile

# common

C = gcc -Wall -Wextra -Wpedantic -fsanitize=address,undefined

A = yasm -f elf64

# compile-only args

CO = -c

S = src/

O = obj/

OBJS = $(O)main.o $(O)afunc.o

all: $(OBJS)

$(C) $(OBJS) -fno-pie -no-pie -o L5

$(O)main.o: $(S)main.c

$(C) $(CO) $(S)main.c -o $(O)main.o

$(O)afunc.o: $(S)afunc.asm

$(A) $(S)afunc.asm -o $(O)afunc.o

clean:

rm $(O)\*.o && rm L5

3. Сборка проекта

make

4. Выполнение программы

4.1. Запуск программы

./L5

4.2. Входные данные

-1753.2753 27.7521195 1000

4.3. Ожидаемый результат выполнения

S pomoshyu ASM:

Num = 2000.1, den = 1755.5, res = 1.1

S pomoshyu C:

Num = 2000.1, den = 1755.5, res = 1.1

4.4. Результат выполнения

S pomoshyu ASM:

Num = 2000.1, den = 1755.5, res = 1.1

S pomoshyu C:

Num = 2000.1, den = 1755.5, res = 1.1

5. Вывод

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