ГУАП

КАФЕДРА № 34

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | Жиданов К.А. |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
|  |
| по курсу: |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 3145 |  |  |  | Калинин Е.Д. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2021

**Задание 1:** написать на языке ассемблера программу, реализующую вычисления в целых числах по заданной формуле. Проверить корректность работы программы с помощью отладчика на заданных значениях.

**Ход выполнения:**

1. Индивидуальное условие задания:

**Выражение**: a/b-c

**Значения**: {31,19,9}

**Разрядность** **входных** **значений**: 8

**Код**: дополнительный

3110  = 1F16

1910 = 1316

910 = 916

1. Регистры для входных значений:

**Входные значения**

a – AL

b – AH

c – BL

1. Заносим программу в отладчик и проверяем значения во всех используемых регистрах
2. Проверяем результаты программы:

**Код программы:**

mov ebp, esp; for correct debugging

;write your code here

mov al, 0x1F

mov ah, 0x13

div al

mov bl, 0x9

sub al, bl

ret

1. Индивидуальное условие задания:

**Выражение**: a/b-c

**Значения**: {-20,-11,2}

**Разрядность** **входных** **значений**: 8

**Код**: дополнительный

-2010  = EC16

-1110 = F516

210 = 216

1. Регистры для входных значений:

**Входные значения**

a – AL

b – AH

c – BL

1. Заносим программу в отладчик и проверяем значения во всех используемых регистрах
2. Проверяем результаты программы:

**Код программы:**

mov ebp, esp; for correct debugging

;write your code here

mov al, 0xEC

mov ah, 0xF5

xor ah, ah

idiv al

mov bl, 0x2

sub al, bl

ret

**Задание 2:** написать на языке ассемблера программу, реализующую вычисления в числах с плавающей точкой по заданной формуле. Проверить корректность работы программы с помощью отладчика на заданных значениях

**Ход выполнения:**

1. Индивидуальное условие задания:

**Выражение**: a/b-c

**Значения**: {31,19,9} {-20,-11,2}

**Разрядность** **входных** **значений**: 8

**Код**: дополнительный

1. Вычислим значения выражения для заданных значений:

Для первого набора данных:

31=41C57000

19=41858000

9=41008000

Для второго набора данных:

-20=C1B20000

-11=C1058000

2=40000000

1. Переменные, используемые в section. data для ввода и вывода данных:

Ввод данных (1): Ввод данных (2):

a: 0x41C57000 a: 0xC1B20000

b: 0x41858000 b: 0xC1058000

c: 0x41008000 c: 0x40000000

1. Заносим программу в отладчик и проверяем значения во всех используемых регистрах
2. Проверяем результаты программы:

**Код программы (1):**

mov ebp, esp; for correct debugging

;write your code here

fld dword [a]

fld dword [b]

fdiv

fld dword [c]

fsub

ret

;section.data

a: dd 0x41C57000

b: dd 0x41858000

c: dd 0x41008000

**Код программы (2):**

xor eax, eax

fld dword [a]

fld dword [b]

fdiv

fld dword [c]

fsub

ret

section.data

a: dd 0xC1B20000

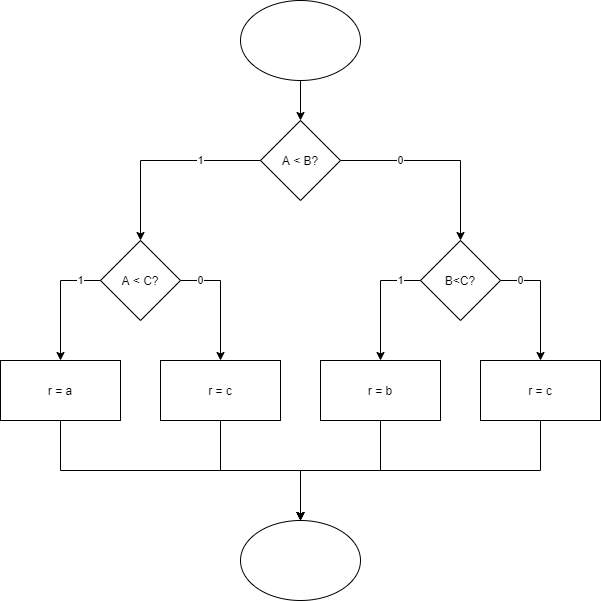
b: dd 0xC1058000

c: dd 0x40000000

**Задание 3:**

Вычислить наибольшее расстояние между тремя точками a,b и c на числовой оси.

**Ход работы:**

1. алгоритм решения задачи:
2. Заносим программу в отладчик
3. Подаём на вход тестовые наборы, программа работает корректно, полученные данные совпадают с теми, что были получены вручную

**Код программы:**

%include "io.inc"

;x-длина первого числового отрезка

;y-длина второго числового отрезка

;z-длина третьего числового отрезка

;section.data

a: dd 0x00000001

b: dd 0x00000002

c: dd 0x00000003

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

;write your code here

xor eax, eax

mov ax, [a] ; загружаем в регистр ax переменную a

mov bx, [b] ; загружаем в регистр bx переменную b

cmp ax, bx ; сравниваем a с b

jg Obhod1 ; если a>b то переход на Obhod1

; иначе в ax где будем максимум грузим BX

mov ax, bx

Obhod1:

mov bx, [c] ; загружаем в регистр bx переменную c

cmp ax, bx ; сравниваем с c

jg Obhod2 ; если > то переход на Obhod2

; иначе в ax где будем максимум грузим BX

mov ax, bx

Obhod2:

; теперь в ax максимум из 3 переменных в памяти: a, b, c

ret