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 |  |  |  | Мухаметшин А. М. |
|  |  |  | 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**: [{28,21,17}, {-15,12,-22}]

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

**Код**: дополнительный

1. Вычислим значения выражения для заданных значений:

Для первого набора данных: 28\*(21+17) =1064

Для второго набора данных: -15\*(12-22) =150

1. Перевод всех чисел в 16-ричную систему с учётом разрядности и способа кодирования:

Первый набор данных: Второй набор данных:

2810  = 1C16 -1510 = FFF116

2110 = 1516 1210 = C16

1710 = 1116 -2210 = FFEA16

1. Регистры для входных и выходных значений:

**Входные значения**: **Выходные значения**:

a – AL r – AL, AX (если результат не помещается в AL)

b – BL

c – CL

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

Первый набор данных:

После выполнения программы в регистре AX находится число 428, что соответствует 1064.

Второй набор данных:

После выполнения программы в регистре AL находится число 96, что соответствует 150

Данные значения совпадают со значениями, полученными вручную.

**Код программы:**

%include "io.inc"

; a\*(b+c)

; 28 21 17

; -15 12 -22

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

;write your code here

MOV AL, 0x1C ;0xFFF1

MOV BL, 0x15;0xC

MOV CL, 0x11 ;0xFFEA

ADD BL, CL

MUL BL

PRINT\_UDEC 1,AX ;AL

xor eax, eax

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**: [{28,21,17}, {-15,12,-22}]

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

**Код**: дополнительный

1. Вычислим значения выражения для заданных значений:

Для первого набора данных: 28\*(21+17) =1064

Для второго набора данных: -15\*(12-22) =150

1. Перевод всех чисел в формат чисел с плавающей точкой в соответствии с форматом IEEE 754

Первый набор данных: Второй набор данных:

2810  = 41E00000 -1510 = C1700000

2110 = 41A80000 1210 = 41400000

1710 = 41880000 -2210 = C1B00000

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

Ввод данных: Вывод данных:

a: a r: r

b: b

c: c

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

Первый набор данных:

После выполнения программы переменной r присваивается значение 44850000, которое при обратном переводе даёт 1064

Второй набор данных:

После выполнения программы переменной r присваивается значение 43160000, которое при обратном переводе даёт 150

Данные значения совпадают со значениями, полученными вручную

**Код программы:**

%include "io.inc"

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

;write your code here

xor eax, eax

FLD DWORD [b]

FLD DWORD [c]

FADD

FLD DWORD [a]

FMUL

FST DWORD [r]

ret

section .data

a: DD 0xC1700000 ;0x41E00000

b: DD 0x41400000 ;0x41A80000

c: DD 0xC1B00000 ;0x41880000

r: DD 0x00000000

**Задание 3:**

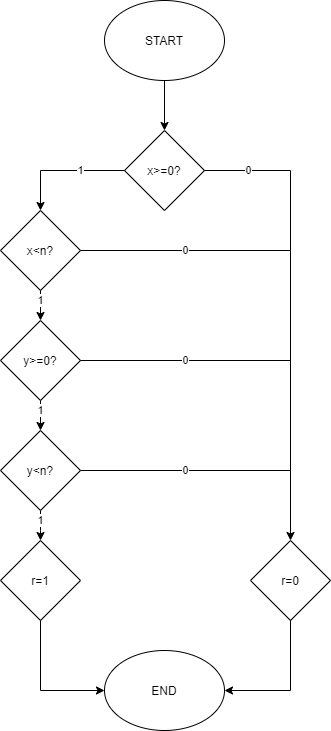
Написать на языке ассемблера программу, реализующую сравнение целых чисел с учётом способа кодирования. Проверить корректность работы программы с помощью отладчика на заданных значениях.

**Ход работы:**

1. Указать попадает ли точка с координатами x;y в квадрат с координатами углов (0;0) (0;n) (n;n) (n;0).

1. Составим алгоритм решения задачи:

В связи с не пониманием, как сделать задание с учётом 3 четверти (отрицательные координаты), сделал только для 1.



1. Проверим корректность работы алгоритма на тестовых наборах данных

С тестовыми значениями алгоритм работает корректно

1. Регистры для входных и выходных значений:

**Входные значения**: **Выходные значения**:

a – AL r – DL

b – BL

c – CL

1. Заносим программу в отладчик
2. Подаём на вход тестовые наборы, программа работает корректно, полученные данные совпадают с теми, что были получены вручную
3. На первом наборе данных (2;2;6) программа выдаёт значение 1, на втором наборе данных (6;6;2) – 0.

**Код программы:**

%include "io.inc"

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

;write your code here

xor eax, eax

mov ax, 2

mov bx, 2

mov cx, 6

cmp ax, 0

jl false

cmp ax, cx

jnl false

cmp bx, 0

jl false

cmp bx, cx

jnl false

true:

mov dx, 1

jmp quit

false:

xor dx,dx

quit:

PRINT\_DEC 1, DX

ret