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 |  |  |  | Одинцов А.А. |
|  |  |  | 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**: [{31,19,13}, {12,-3,-10}]

**Разрядность** **входных** **значений**: 16

**Код**: дополнительный

1. Вычислим значения выражения для заданных значений:

Для первого набора данных: (31+19) \* 13 = 650

Для второго набора данных: (12 - 3) \* (-10) = -90

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

Первый набор данных: Второй набор данных:

3110 = 1F 16  1210 = C16

1910 = 1316 -310 = FD16

1310 = D16  -1010 = F616

1. Регистры для входных и выходных значений:

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

a – AX r - EAX

b – BX

c – CX

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

После выполнения программы в регистре EAX находится число 28A16, что соответствует числу 65010

Данные значения совпадают со значениями, полученными вручную.

Код программы на языке ассемблера:

%include "io.inc"

;(a+b)\*c

;31 19 13

;12 -3 -10

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

MOV AX, 0x1F ;0xC

MOV BX, 0x13 ;0xFFFD

MOV CX, 0xD ;0xFFF6

ADD AX, BX

IMUL CX

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**: [{31,19,13}, {12,-3,-10}]

**Разрядность** **входных** **значений**: 16

**Код**: дополнительный

1. Вычислим значения выражения для заданных значений:

Для первого набора данных: (31+19) \* 13 = 650

Для второго набора данных: (12 - 3) \* (-10) = -90

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

Первый набор данных: Второй набор данных:

3110 =41f80000 1210 = 41400000

1910 = 41980000 -310 = c0400000

1310 = 41500000 -1010 = c1200000

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 0x44228000, которое при обратном переводе даёт 65010

Второй набор данных:

После выполнения программы переменной r присваивается значение 0xc2b40000, которое при обратном переводе даёт -9010

Данные значения совпадают со значениями, полученными вручную

%include "io.inc"

;(a+b)\*c

;31 19 13

;12 -3 -10

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

xor eax, eax

FLD DWORD [c]

FLD DWORD [a]

FLD DWORD [b]

FADD

FMUL

FST DWORD [r]

ret

section .data

a: DD 0x41f80000 ;0x41400000

b: DD 0x41980000 ;0xc0400000

c: DD 0x41500000 ;0xc1200000

r: DD 0x00000000

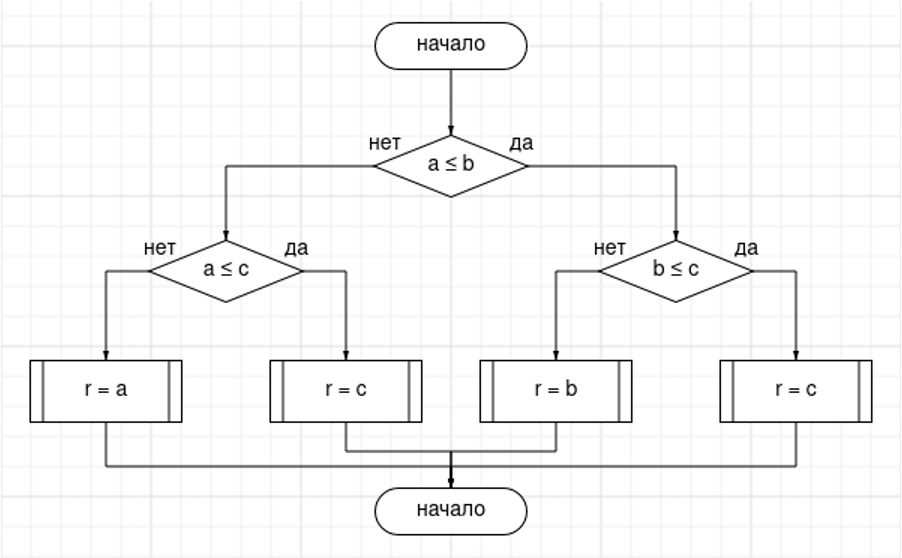
**Задание 3.** Написать на языке ассемблера программу, реализующую сравнение целых чисел с учетом способа кодирования. Проверить корректность работы программы с помощью отладчика на заданных значениях. Нахождение максимального из трех значений.

**Ход выполнения задания:**

1. Индивидуальное условие задания  
   **Значения:** [{31, 19, 13};{12, -3, -10}]
2. На данных наборах:  
   13<19<31  
   -10<-3<12
3. Перевод всех используемых чисел в 16-ричную систему с учетом разрядности и способа кодирования прямой/дополнительный код Первый набор данных: Второй набор данных:  
    3110 = 1F 16  1210 = C16

1910 = 1316 -310 = FD16

1310 = D16  -1010 = F616Отрицательные числа представляются в дополнительном коде с помощью инверсирования двоичных представлений равных им по модулю чисел и увеличения этих представлений на 1

1. Регистры ля хранения входных и выходных значений:  
    Входные значения Выходные значения  
     
    a: AX maximum: DX  
    b: BX  
    c: CX
2. Алгоритм работы программы на языке ассемблера
3. Отладка программы с отслеживанием содержимого регистров EAX, EBX, ECX
4. Результат работы программы совпадает с вычисленным вручную для первого из текстовых наборов

%include "io.inc"

section .text

global CMAIN

CMAIN:

mov ebp, esp; for correct debugging

xor eax, eax

xor ebx, ebx

xor ecx, ecx

xor edx, edx

mov al, 0x1F ; 0xC

mov bl, 0x13 ; 0xFFFD

mov cl, 0xD ; 0xFFF6

cmp ax, bx

jnl num\_1

cmp bl, cl

jnl num\_2

mov dl, cl

jmp quit

num\_2:

mov dl, bl

jmp quit

num\_1:

cmp al, cl

jnl num\_3

mov dl, cl

jmp quit

num\_3:

mov dl, al

quit:

ret