**Практическая работа № 7**

|  |  |  |  |
| --- | --- | --- | --- |
| Выполнил | Студент гр. МФ-11 | Лазаренко А. В. | 04.04.2021 |
| Проверил | Ст. преподаватель | Меняйлов Е. С. |  |

**Тема работы:**

«Вычисление арифметических выражений»

**Цель работы:**

Вычислить заданное смешанное арифметическое

выражение для данных в заданных форматах.

**Задание:**

1. Проанализировать арифметическое выражение, определить

порядок выполнения операций, типы получающихся значений, там,

где возможно, упростить.

2. Вычислить область определения выражения

3. Оценить значения переменных, при которых не произойдет

переполнения.

4. Составить программу на Ассемблере для вычисления выражения.

**Вариант 11:**

**Номер 1**

Variables ‘a’ and ‘c’ are 8-bit, so their maximum values are [-128; 127]. Variables ‘b’ and ‘d’ are 16-bit, so their maximum values are [-65536; 65535]. Otherwise, calculations will not be correct. Also, they will not be correct if the value of ‘a’ in a/4 will not be aliquot to 4, and if the value of ‘a’ = 4’d’, the denominator = 0, which cannot be calculated.

/\* Algorithm:

\* 1) declaring necessary variables a, b, c, d, which are being used in a mathematical expression;

\* 2) declaring necessary variables whole\_result, remainder\_result for storing the whole part and the quotient;

\* 3) converting a\_temp, c\_temp to char and initialising variables a and c with them;

\* 4) moving value of a variable a to al register;

\* 5) shifting bits of the number of the number stored in al register two bits right (dividing by 4);

\* 6) expand al register to ax register (adding 8 zeros from the left side);

\* 7) moving value of a variable d to bx register;

\* 8) subtracting ax and bx registers, result stored in ax register;

\* 9) moving value of register ax a to register cx;

\* 10) moving value of c variable a to al register;

\* 11) shifting bits of the number of the number stored in al register one bit left (multiplying by 2);

\* 12) expand ax register to the pair of registers ax:dx;

\* 13) moving value of b variable a to bx register;

\* 14) multiplying the value stored in bx register by 23;

\* 15) adding ax and bx registers, result stored in ax register;

\* 16) subtracting ax register and 12, result stored in ax register;

\* 17) dividing a pair of register ax:dx by the value in register cx -> whole part stored in ax, fractional in dx;

\* 18) putting the value stored in ax register to the variable whole\_result;

\* 19) putting the value stored in dx register to the variable remainder\_result;

\* 20) outputting the result values to the screen.

\*/

#include <iostream>

using namespace std;

int main(void)

{

short a\_temp, b, c\_temp, d;

cin >> a\_temp >> b >> c\_temp >> d;

char a = char(a\_temp);

char c = char(c\_temp);

int temp\_result;

int whole\_result, remainder\_result;

\_\_asm // (2с + 23b - 12)/(a/4 - d)

{

// знаменатель

mov al, a;

sar al, 2;

cbw;

mov bx, d;

sub ax, bx;

mov cx, ax;

movsx ecx, cx;

// числитель

mov ax, b;

mov bx, 23;

imul bx;

shl edx, 16;

mov dx, ax;

mov ebx, edx;

mov al, c;

cbw;

sal ax, 1;

movsx eax, ax;

add eax, ebx;

sub eax, 12;

cdq;

// итог

idiv ecx;

mov whole\_result, eax;

mov remainder\_result, edx;

}

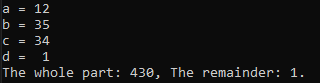
printf("a = %d\nb = %d\nc = %d\nd = %2d", a\_temp, b, c\_temp, d);

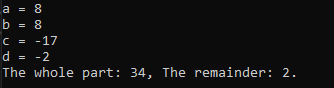
printf("\nThe whole part: %d, The remainder: %u.\n", whole\_result, remainder\_result);

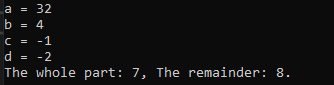
return 0;

}

**Результаты выполнения программы:**

1) 

2) 

3) 

**Задание \***

#include <cmath>

#include <iostream>

using namespace std;

int main(void)

{

float a = 12.32;

float b = 35.129;

float c = 71.908;

float d = 0.01;

float x = 4;

float y = 2;

float z = 23;

float u = 12;

float result = 0;

\_\_asm

{

fld a; // передаю в st(0) значение a

fdiv x; // делю st(0) = a на 4

fld d; // передаю в st(0) значение d

fxch st(1); обмениваю значения 0 и 1 регистров

fsub st(0), st(1); // вычитаю из 0 регистра 1, результат записывается в st(0)

fld c; // передаю в st(0) значение c

fmul y; // умножаю st(0) на значение y

fld b; // передаю в st(0) значение b

fmul z; // умножаю st(0) на значение z

fadd st(0), st(1); // складываю 0 и 1 регистры, результат записывается в st(0)

fld u; // передаю в st(0) значение u

fsub st(1), st(0); // вычитаю из 1 регистра 0, результат записывается в st(1)

fxch st(3); // обмениваю значения 0 и 3 регистров

fst result; // значение result = st(0) (бывшему st(3))

fxch st(1); // обмениваю значения 3 и 0 регистров обратно

fdiv result; // делю числитель в st(0) на знаменатель = result

fst result; // значение result = st(0) – итоговому значению выражения

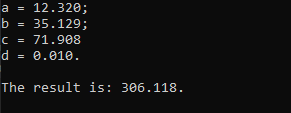
}

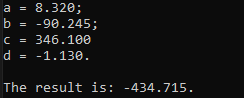
printf("\n%.3f.\n", result);

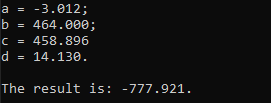
return 0;

}

**Результаты выполнения программы:**

1) 

2) 

3) 

fld – команда передачи в st(0) значения источника;

fst – команда передачи из регистра st(0) в указанный операнд;

fadd – команда сложения (регистров) первого операнда со вторым, результат записывается в первый операнд;

fsub – команда вычитания (регистров) вторпого операнда из первого, результат записывается в первый операнд;

fmul – команда умножения (регистра st(0) по умолчанию) на значение источника;

fdiv – команда деления (регистра st(0) по умолчанию) на значение источника;

fxch – команда обмена значениями двух регистров (если указан только один операнд, меняет значения регистров st(i) с st(0)).

**Вывод**

Я смог вычислить заданное смешанное арифметическое выражение для данных в заданных форматах, используя язык Ассемблер и дополнение арифметического сопроцессора.