

ЛАБОРАТОРНАЯ РАБОТА №2

по дисциплине "Языки Ассемблера"

Выполнил Ковалев Даниил
СКБ171, вариант 12
МИЭМ НИУ ВШЭ

Дата: / / 2020

Баллы:

Задание: $v = \frac{2x + (z + 4)y}{3x - 1} - 2$, x — байт, y, z, v — слова.

1) $x = 1h, y = -2h, z = -6h, v = 1h$; 2) $x = -6Bh, y = 5F3h, z = 12h, v = -69h$;

```

1  #include <iostream>
2  #include <iomanip>
3  #include <cstdint>
4
5  using BYTE = std::int8_t;
6  using WORD = std::int16_t;
7
8  void lab(BYTE x, WORD y, WORD z)
9  {
10     std::cout << std::hex << std::internal << std::setfill('0') << std::showbase
11         << "x = " << std::setw(4) << (0xff & static_cast<int>(x)) << ", "
12         << "y = " << std::setw(6) << y << ", "
13         << "z = " << std::setw(6) << z << '\n';
14     WORD v_c = (2 * x + (z + 4) * y) / (3 * x - 1) - 2;
15     std::cout << std::hex;
16     std::cout << "Result (C computed): " << std::setw(6) << v_c << '\n';
17
18     WORD v_asm;
19     asm( ".intel_syntax\n\t"
20         "mov ax, dx\n\t" // z->AX
21         "add ax, 4\n\t" // (z+4)->AX
22         "imul cx\n\t" // (z+4)*y->DX:AX
23         "mov cx, ax\n\t" // (z+4)*y->DX:CX
24         "mov al, bl\n\t" // x->AL
25         "mov bx, dx\n\t" // (z+4)*y->BX:CX
26         "cbw\n\t" // [BYTE]x->[WORD]x (AX)
27         "sal ax, 1\n\t" // 2*x->AX
28         "cwd\n\t" // [WORD]2*x->[DWORD]DX:AX
29         "add cx, ax\n\t"
30         "adc bx, dx\n\t" // 2*x+(z+4)*y->BX:CX
31         "sar ax, 1\n\t" // x->AX
32         "mov dl, 3\n\t" // 3->DL
33         "imul dl\n\t" // 3*x->AX
34         "sub ax, 1\n\t" // (3*x-1)->AX
35         "mov dx, bx\n\t" // 2*x+(z+4)*y->DX:CX
36         "mov bx, ax\n\t" // (3*x-1)->BX
37         "mov ax, cx\n\t" // 2*x+(z+4)*y->DX:AX
38         "idiv bx\n\t" // (2*x+(z+4)*y)/(3*x-1)->AX
39         "sub ax, 2\n\t" // (2*x+(z+4)*y)/(3*x-1)-2->AX
40         : "=a"(v_asm)
41         : "b"(x), "c"(y), "d"(z)
42     );
43     std::cout << "Result (ASM computed): " << std::setw(6) << v_asm << '\n';
44 }
45
46 int main()
47 {
48     BYTE x1, x2;
49     WORD y1, y2, z1, z2, v1, v2;
50
51     x1 = 0x01, y1 = -0x0002, z1 = -0x0006, v1 = 0x0001,
52     x2 = -0x6b, y2 = 0x05f3, z2 = 0x0012, v2 = -0x0069;
53     std::cout << std::hex << std::internal << std::setfill('0') << std::showbase;
54
55     lab(x1, y1, z1);
56     std::cout << "Expected result: " << std::setw(6) << v1 << "\n\n";
57     lab(x2, y2, z2);
58     std::cout << "Expected result: " << std::setw(6) << v2 << "\n\n";
59
60     return 0;
61 }

```

Результат работы программы:

```
1 x = 0x01, y = 0xfffe, z = 0xfffa
2 Result (C computed): 0x0001
3 Result (ASM computed): 0x0001
4 Expected result: 0x0001
5
6 x = 0x95, y = 0x05f3, z = 0x0012
7 Result (C computed): 0xff97
8 Result (ASM computed): 0xff97
9 Expected result: 0xff97
```

1) Результат **0x0001** совпадает с ожидаемым результатом $v = 1h$.

2) Результат **0xFF97** есть дополнительный код ожидаемого числа $v = -69h = -(6 \cdot 16^1 + 9 \cdot 16^0) = -105_{10}$. Проверим это: $FF97_{16} = 1111\ 1111\ 1001\ 0111_2$,
 $v = -2^{15} + 2^{14} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^9 + 2^8 + 2^7 + 2^4 + 2^2 + 2^1 + 2^0 = -105_{10}$