Правительство Российской Федерации

Федеральное государственное автономное образовательное учреждение

Высшего профессионального образования

Национальный исследовательский институт

**«Высшая школа экономики»**

Московский институт электроники и математики

Прикладная математика

**Отчет**

**По контрольной работе №2**

**По курсу «Язык ассемблера»**

**Вариант №17**

|  |  |  |  |
| --- | --- | --- | --- |
| Ф.И.О. студента | Номер группы | Дата | Баллы |
| Мазур Дарья Александровна | БПМ-185 |  |  |

**Задание А2**

, y, v-слова

х, z-байты

Написать программу, которая вычисляет значение выражения с помощью стандартных арифметических операций языка С и с помощью ассемблерной вставки. Проверить на двух тестовых наборах.

1. x = -1h; y = 2h; z = -1h; v = -3h;
2. x = 6Dh; y = 7FC3h; z = -7Fh; v = 6BF4h;

Код программы

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <locale.h>

#include <string.h>

short int assembler(char x,short int y,char z)

{

short int v\_as;

\_\_asm

{

// считаем числитель

mov al, x;

cbw;

mov bx, y;

add bx, 2;

imul bx;

mov cx, dx;

mov bx, ax;

sub bx, 2;

sbb cx, dx;

// считаем знаменатель

mov al, z;

cbw;

sub ax, 2;

xchg ax, bx;

idiv bx;

neg ax;

sub ax, 1;

mov v\_as, ax;

}

return v\_as;

}

int main()

{

char x, z;

short int y, v,v\_as,v\_c;

setlocale(LC\_ALL, "Rus");

int i;

setlocale(LC\_ALL, "Rus");

printf("Input the number of the test: ");

scanf\_s("%d", &i);

switch (i)

{

case 1:

{

x = -0x1;

y = 0x2;

z = -0x1;

v = -0x3;

break;

}

case 2:

{

x = 0x6D;

y = 0x7FC3;

z = -0x7F;

v = 0x6BF4;

break;

}

default:

printf("The number is wrong\n");

break;

}

v\_c = (2 - x \* (y + 2)) / (z - 2) - 1;

v\_as = assembler(x, y, z);

printf("\nРасчет(Ассемблер):в 10: %d и в 16 %x", v\_as, v\_as);

printf("\nИстинный ответ: в 10 %d и в 16 %x \n", v, v);

printf("\nРасчет(СИ):в 10: %d и в 16 %x", v\_c, v\_c);

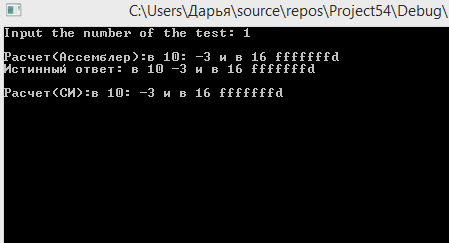
getch();

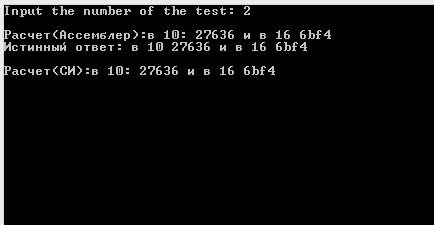
return 0;

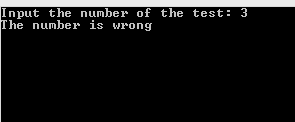
system("pause");

}

**ТЕСТЫ.**

****





Переведем число FFFFFFFD из дополнительного кода в прямо, чтобы убедиться, что оно равно данному значению(-3h):

FFFFFFFD→1111 1111 1111 1111 1111 1111 1111 1101→1111 1111 1111 1111 1111 1111 1111 1100 →

0011→-3h