



Национальный исследовательский университет «Высшая школа экономики»

Дисциплина «»

Специальность 10.05.01 «Компьютерная безопасность»

Отчет
По лабораторной работе №5
«»
по дисциплине «Языки ассемблера»
направления «Компьютерная безопасность»

Вариант 5

Винников Н. В. СКБ-181

Москва, 2021

Виппиков Н.

5. В массиве слов (байтов) выбранного размера (5-6 элементов) выполнить над каждым элементом операцию: если биты 6:4 являются инверсией битов 3:1, то установить биты второго поля и сбросить старший бит. Вывод на экран исходного массива и массива результатов производить в двоичном и шестнадцатеричном представлениях.

```
#include <stdio.h>
// #include <string.h>
#include <locale.h>
#include <stdlib.h>
// #include <ctype.h>
// #include <cstdlib>
// #include <iostream>

void itoa2b(unsigned int a, char * buf)
{
    _asm
    {
        mov     edi,buf
        mov     edx,a
        mov     ecx,8
        m1:
        mov     al,'0'
        shl     dl,1
        adc     al,0
        stosb
        loop    m1
        xor     eax,eax
        stosb
    }
}

void itoa2w(unsigned int a, char* buf)
{
    _asm
    {
        mov     edi, buf
        mov     edx, a
        mov     ecx, 16
        m1:
        mov     al, '0'
        shl     dx, 1
        adc     al, 0
        stosb
        loop    m1
        xor     eax, eax
        stosb
    }
}

int main()
{
    setlocale(LC_ALL, "ru");
    setlocale(LC_NUMERIC, "en");
    char bb[33];
    char bw[33];
    unsigned char ab[6] = {255,0,226,198,156,123 };
    unsigned short aw[6] = { 0xB039, 62512, 0xF0E2, 0xEEC6, 0xCA9C, 54864 };

    printf("Исходные массивы (байты/слова):\n");
    for (unsigned int i = 0; i < 6; i++)
```

```

    {
        itoa2b((unsigned int)ab[i], bb);
        itoa2w((unsigned int)aw[i], bw);
        printf("байт.%.2d=%s/%.2X\tслово.%.2d=%s/%.4X\n", i, bb, (unsigned
int)ab[i], i, bw, (unsigned int)aw[i]);
    }
    // Расчет для байта
    _asm {
        lea        esi,ab // esi - указатель на массив ab
        lea        edi,aw // esi - указатель на массив aw
        mov        ecx,6 // ecx - длина массива
    m1:
    // если биты 6:4 являются инверсией битов 3:1, то установить биты второго поля сбросить
старший бит
        mov        al,[esi] // al - очередной элемент массива ab
        mov        ah,al // ah = al
        shr        al,4 // al >> 4 - переносим биты 6:4 в младшие,
остальные зачищаем
        and        al,7 // Зачищаем у al все биты кроме трех младших
        shr        ah,1 // ah >> 1 - переносим биты на 1 разряд в право
        and        ah,7 // Зачищаем у ah все биты кроме трех младших
        xor        ah,7 // Инвертируем младшие биты ah
        cmp        al,ah // Сравниваем al и ah
        jnz        m2 // Не совпадают
    // Совпадают
        or         byte ptr [esi],4 // устанавливаем бит второго поля. Считаем
от 0
        and        byte ptr[esi],0x7F // сбрасываем старший бит
    m2:

        mov        ax, [edi] // ax - очередной элемент массива aw
        mov        ah, al // ah = al
        shr        al, 4 // al >> 4 - переносим биты 6:4 в младшие,
остальные зачищаем
        and        al, 7 // Зачищаем у al все биты кроме трех младших
        shr        ah, 1 // ah >> 1 - переносим биты на 1 разряд в право
        and        ah, 7 // Зачищаем у ah все биты кроме трех младших
        xor        ah, 7 // Инвертируем младшие биты ah
        cmp        al, ah // Сравниваем с эталоном
        jnz        m3 // Не совпадают
        or byte ptr[edi], 4 // устанавливаем бит второго поля. Считаем от 0
        and byte ptr[edi+1], 0x7F // сбрасываем старший бит
    m3:
        inc        esi // указатель на следующий элемент массива ab
        add        edi,2 // указатель на следующий элемент массива aw
        loop    m1 // цикл
    }

    printf("Результат (байты/слова):\n");
    for (unsigned int i = 0; i < 6; i++)
    {
        itoa2b((unsigned int)ab[i], bb);
        itoa2w((unsigned int)aw[i], bw);
        printf("байт.%.2d=%s/%.2X\tслово.%.2d=%s/%.4X\n", i, bb, (unsigned
int)ab[i], i, bw, (unsigned int)aw[i]);
    }
    system("pause");
}

```

Исходные массивы (байты/слова):

байт.00=11111111/FF	слово.00=1011000000111001/B039
байт.01=00000000/00	слово.01=1111010000110000/F430
байт.02=11100010/E2	слово.02=1111000011100010/F0E2
байт.03=11000110/C6	слово.03=1110111011000110/EEC6
байт.04=10011100/9C	слово.04=1100101010011100/CA9C
байт.05=01111011/7B	слово.05=1101011001010000/D650

Результат (байты/слова):

байт.00=11111111/FF	слово.00=0011000000111101/303D
байт.01=00000000/00	слово.01=1111010000110000/F430
байт.02=01100110/66	слово.02=0111000011100110/70E6
байт.03=01000110/46	слово.03=0110111011000110/6EC6
байт.04=00011100/1C	слово.04=0100101010011100/4A9C
байт.05=01111011/7B	слово.05=1101011001010000/D650

Для продолжения нажмите любую клавишу . . .