

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Организация ЭВМ и систем»
Тема: Представление и обработка символьной информации с
использованием строковых команд.

Студентка гр. 1381

Манцева Т.К.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Изучить обработку символьной информации с использованием строковых команд.

Задание.

Разработать программу обработки символьной информации, реализующую функции:

- инициализация (вывод титульной таблички с указанием вида преобразования и автора программы) - на ЯВУ;
- ввода строки символов, длиной не более N_{\max} (≤ 80), с клавиатуры в заданную область памяти - на ЯВУ; если длина строки превышает N_{\max} , остальные символы следует игнорировать;
- выполнение заданного в таблице 5 преобразования исходной строки с записью результата в выходную строку - на Ассемблере;
- вывода результирующей строки символов на экран и ее запись в файл - на ЯВУ.

Ассемблерную часть программы включить в программу на ЯВУ по принципу встраивания (in-line).

Вариант 10

Преобразование введенных во входной строке шестнадцатеричных цифр в двоичную СС, остальные символы входной строки передаются в выходную строку непосредственно.

Выполнение работы.

Была разработана программа на языке Си с ассемблерной вставкой. Во время выполнения кода на Ассемблере для каждого символа выполняется проверка, входит ли код текущего символа в диапазон кодов шестнадцатеричных цифр 0-9, A-F. Если его код принадлежит этим диапазонам, то символ преобразовывается в число, далее постепенно сравнивается со степенями двойки и разбивается на них, его двоичный код записывается в строку-результат. В противном случае он сохраняется в

выходную строку без преобразований. Чтение и запись символов осуществлены с помощью команд `lods b`, `stos b` соответственно. Разработанный код представлен в приложении 1.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	qwertyABCDEF123456789	qwerty101010111100110111101111000100100011010001010110011110001001	Программа работает правильно
2.	AasdfB	1010asdf1011	Программа работает правильно
3.	123tu tu ru	000100100011tu tu ru	Программа работает правильно
4.	nothing	nothing	Программа работает правильно

Выводы.

Были изучены представление и обработка символьной информации с использованием строковых команд.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: lab4.c

```
#include <stdio.h>
#define SIZE 81

int main(){

printf("Mantseva Tatyana, group 1381. From hexadecimal to binary
numbers\n");

char input[SIZE];
char result[SIZE*4];

fgets(input,SIZE,stdin);

asm ( "check:          \n"
      "\t lodsrb        \n"
      "\t cmp %%al,0      \n"
      "\t je exit         \n"
      "\t cmp %%al,'0'     \n"
      "\t jl write        \n"
      "\t cmp %%al,'F'     \n"
      "\t jg write        \n"
      "\t cmp %%al,'9'     \n"
      "\t jle number      \n"
      "\t cmp %%al,'A'     \n"
      "\t jge letter      \n"

      "\t number:        \n"
      "\t sub %%al,48      \n"
      "\t jmp binary      \n"

      "\t letter:        \n"
      "\t sub %%al,55      \n"

      "\t binary:        \n"
      "\t mov %%bl,%%al     \n"
      "\t cmp %%bl, 8      \n"
      "\t mov %%al,'0'     \n"
      "\t jl fourth        \n"
      "\t mov %%al,'1'     \n"
      "\t sub %%bl, 8      \n"

      "\t fourth:        \n"
      "\t stosb            \n"
      "\t cmp %%bl,4       \n"
      "\t mov %%al,'0'     \n"
      "\t jl third         \n"
      "\t mov %%al,'1'     \n"
      "\t sub %%bl, 4       \n"
```

```

        "\t third:          \n"
        "\t stosb           \n"
        "\t cmp %%bl,2       \n"
        "\t mov %%al,'0'      \n"
        "\t jl second         \n"
        "\t mov %%al,'1'      \n"
        "\t sub %%bl, 2       \n"

        "\t second:        \n"
        "\t stosb           \n"
        "\t cmp %%bl,1        \n"
        "\t mov %%al,'0'      \n"
        "\t jl first          \n"
        "\t mov %%al,'1'      \n"
        "\t sub %%bl, 1       \n"

        "\t first:         \n"
        "\t stosb           \n"
        "\t jmp end           \n"
        "\t write:          \n"
        "\t stosb           \n"
        "\t end:            \n"
        "\t jmp check       \n"
        "\t exit:          \n"
        : "=m" (result)
        : "S" (input),
        : "D" (result)
    );

    printf("result = %s", result);
    return 0;
}

```