

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.

Студент гр. 0383

Пенкин М.В.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Цель работы.

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).

Замечания:

- 1) При выполнении преобразования обязательно использовать команды работы со строками;
- 2) При выполнении преобразования нельзя портить входную строку. Результат преобразования должен записываться в выходную строку.

Вариант 8:

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

Выполнение работы.

Для выполнения данной лабораторной работы был выбран язык C++ и среда разработки Visual Studio. Блок ассемблерного кода вставлен в программу после ключевого слова `__asm`. В программе происходит считывание строки длиной не более 80 символов командой `fgets`,

последний символ, который является символом перевода строки, заменяется на завершающий символ (с кодом 0). 3

В ассемблерном блоке происходит обработка введенной строки, в цикле с помощью команды `lodsb` считывается очередной символ введенной строки, далее проверяется является ли данный символ 'А', 'В', 'С', 'D', 'Е', или 'F' (этих цифр нет в десятичной системе счисления) и если он является одной из этих цифр, то в выходную строку с помощью команды `stosw` заносится 2 символа: "10", "11", "12", "13", "14", "15" соответственно, иначе символ записывается в выходную строку неизменным с помощью команды `stosb`. В конце программы полученная выходная строка выводится на экран и в текстовый файл.

Таблица 1. Проверка работы программы

Введенная строка	Результирующая строка	Примечание
123456789ABCDEF	123456789101112131415	Верно
hello i am fine	hello i am fine	Верно
AaBbCcDdEeFfGgHh	10a11b12c13d14e15fGgHh	Верно
AAakekBBB	101010kek111111	Верно
A B C mem C B A 123	10 11 12 mem 12 11 10 123	Верно

Тексты исходных файлов программ см. в приложении А.

Выводы.

В ходе выполнения данной лабораторной работы была изучена обработка символьной информации с помощью строковых команд на языке Ассемблер.

ПРИЛОЖЕНИЕ А

ТЕКСТЫ ИСХОДНЫХ ФАЙЛОВ ПРОГРАММ

Название файла: lr4.cpp

```
#include <iostream>
#include <stdio.h>

char s[81];
char outstr[161];

int main()
{
    fgets(s, 81, stdin);
    s[strlen(s) - 1] = '\\0';
    __asm {
        push ds
        pop es
        mov esi, offset s
        mov edi, offset outstr
    L :
        lodsb; в al очередной символ

        cmp al, 'A'; является ли введённый символ 'A'
        jne skip1
        mov al, '1'
        mov ah, '0'
        stosw
        jmp final
    skip1:

        cmp al, 'B'; является ли введённый символ 'B'
        jne skip2
        mov al, '1'
        mov ah, '1'
        stosw
        jmp final
    skip2:

        cmp al, 'C'; является ли введённый символ 'C'
        jne skip3
        mov al, '1'
        mov ah, '2'
        stosw
        jmp final
    skip3:

        cmp al, 'D'; является ли введённый символ 'D'
        jne skip4
        mov al, '1'
        mov ah, '3'
        stosw
        jmp final
    skip4:
    }
```

```

    cmp al, 'E'; является ли введённый символ 'E'
    jne skip5
    mov al, '1'
    mov ah, '4'
    stosw
    jmp final
skip5:

```

```

    cmp al, 'F'; является ли введённый символ 'F'
    jne skip6
    mov al, '1'
    mov ah, '5'
    stosw
    jmp final
skip6:

```

stosb; кладем в выходную строку байт из al

```

    final:
    mov ecx, '\0'
    cmp ecx, [esi]
    je LExit; выход из цикла, если текущий символ
завершающий
    jmp L
LExit :

```

```

};
std::cout << outstr;
FILE* f;
fopen_s(&f, "out.txt", "w");
fwrite(outstr, sizeof(char), strlen(outstr), f);
return 0;
}

```