

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. 1381

Сагидуллин Э.Р.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2022

Цель работы.

Изучить представление и обработку символьной информации с использованием строковых команд. Изучить понятие ветвящихся процессов и их организацию.

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 19:

Заменить введенные во входной строке латинские буквы на десятичные числа, соответствующие их номеру по алфавиту, остальные символы входной строки передать в выходную строку непосредственно.

Ход выполнения работы

В начале программы подключаются заголовки необходимых библиотек. Выделяется память для входного и выходного массива символов (строк). Далее, выводится специальная строка содержащая задание и авторскую лицензию. Считывается входная строка с ограничением количества символов по условию. Управление передается в вставку с кодом на языке ассемблера. Далее, происходит считывание входной строки по символно, и последующее

распределение символов по меткам, исходя из принадлежности промежуткам `ascii` таблицы. В зависимости от положения символа в таблице, ему сопоставляется необходимый символ с помощью заданного смещения. Далее, итоговый байт записывается в выходную строку. В случае, когда позиция буквы в алфавите состоит из двух цифр используется специальная команда для записи двух байт (слова) в выходную строку. Перемещение по меткам происходит с помощью специальных ключевых слов `jge`, `jl`, `jle` и `jmp`. Происходит выход из вставки и вывод итоговой строки в консоль. Для записи байт в строку использовались команды `stosb` и `stosw`. Также, возможен вывод в файл.

Тестирование

| Входные данные | Результат |
|-------------------------------|--------------------------------------|
| 123 !@#\$ qweQWE ЁЙЦУёйцу | 123 !@#\$ 1723517235 ЁЙЦУёйцу |
| a1b2c3d4e5ABCDE!@#\$\$%^&1234 | 112233445512345!@#\$\$%^&1234 |
| MikhailTheBestPrepod123456 | 139118191220852519201618516154123456 |

Вывод

В ходе выполнения лабораторной работы были изучены представление и обработка символьной информации с использованием строковых команд.

ПРИЛОЖЕНИЕ А

КОД ПРОГРАММЫ

Название файла: lab4.cpp

```
#include <fstream>
#include <cstdio>

char input[81];
char output[162];

int main() {

    std::cout << "Sagidullin Ernest 1381.\nReplace the Latin letters
entered in the input string with decimal numbers, corresponding to their
alphabetical number\n";
    std::cin.getline(input, 81);

    __asm {
        push ds
        pop es
        mov esi, offset input
        mov edi, offset output
        read :
            lodsb
            cmp al, '\0'
            je end

            cmp al, 'A'
            jl write

            cmp al, 'Z'
            jle write2

            cmp al, 'a'
            jl write

            cmp al, 'z'
            jle writel

        write :
            stosb
            jmp read

        writel :
            and al, 5FH

        write2 :
            cmp al, 'J'
            jge write3
            sub al, 'A'
            add al, 31h
            jmp write

        write3 :
            cmp al, 'T'
            jge write4
```

```

        mov ah, al
        sub ah, 'J'
        add ah, 30h
        mov al, '1'
        stosw
        jmp read

write4 :
        mov ah, al
        sub ah, 'T'
        add ah, 30h
        mov al, '2'
        stosw
        jmp read
end :
}

std::cout << output;
std::ofstream file;
file.open("output.txt");
file << output;
file.close();

return 0;
}

```