

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. 0382

Азаров М.С.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

Цель работы.

Научится обработке символьной информации с использованием строковых команд. Также научится встраивать ассемблерный код в код языков высокого уровня по принципу *in-line*.

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 1

Формирование выходной строки только из цифр и русских букв входной строки.

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.

Выполнение работы.

Выделяем память под вводимую и выводимую строку.

```
char input[81];  
char output[81];
```

Функция `main()` выводит приветственное сообщение с автором программы и описанием преобразования которое она делает. Считывает вводимую строку , открывает файл для чтения .

```
cout << "Азаров Максим 0382, формирование выходной строки только из цифр и русских букв входной строки:" << endl;  
ofstream file;  
file.open("res.txt");  
cin.getline(input, 81);
```

Затем идет код на ассемблере встроенный в программу по принципу *in-line* , которые преобразует вводную строку в соответствии с заданием .

Ассемблерная часть программы:

1. Сохраняем в индексные регистры смещение в базе данных для входной и выходной строки.

```
mov esi, offset input  
mov edi, offset output
```

2. Помечаем начало блока обработки текущего символа *check_symbol* и используем команду *lodsb* , которая загружает в регистр AL байт из ячейки

памяти, указываемой при помощи индексного регистра-источника.

```
check_symb1 :  
    lodsb
```

3. Далее с помощью команды сравнения *cmp* и команд условного перехода определяется в каком диапазоне находится текущий символ в кодировке WIN-1251 (текущий символ хранится в AL). Если этот символ находится в диапазоне цифр, переход осуществляется на метку сохранения символа в выходную строку. Аналогично с русскими буквами + ё и Ё. Если текущий символ "\0" программа завершает обработку и выходит из ассемблеровской части.

```
    cmp al, '0'  
    jb check_symb1  
  
    cmp al, '9'  
    jbe write_symb1  
  
    cmp al, 'ё'  
    je write_symb1  
  
    cmp al, 'Ё'  
    je write_symb1  
  
    cmp al, 'А'  
    jb check_symb1
```

4. Блок ответственный за сохранение в выходную строку нужных символов. С помощью команды *stosb* , которая сохраняет регистр AL в ячейке памяти по адресу DS:EDI. И перенаправляет выполнения программы на обработку следующего символа.

```
write_symb1 :  
    stosb  
    jmp check_symb1
```

5. Блок кода выхода из цикла обработки строки , выполняется когда встречен символ "\0". Записывает этот символ в выходную строку с помощью команды *stosb*.

```
finish :  
    stosb
```

После ассемблерной части обработки входной строки , программа выводит выходную строку в консоль и файл и закрывает файл.

```
cout << output;  
file << output;  
file.close();
```

Тестирование.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	ауыыаһufieshsie678	ауыыа678	Программа работает корректно
2.	HSFS^&^*^6786АПУЫ	6786АПУЫ	Программа работает корректно
3.	sheЁё4937ыауАГШ	Ёё4937ыауАГШ	Программа работает корректно

Выводы.

Была изучена обработка символьной информации на языке ассемблер с использованием строковых команд.

В ходе данной лабораторной работы была разработана программа , которая формирует выходную строку только из цифр и русских букв входной строки. Программа написана на языке высокого уровня (C++), с использованием встроенного по принципу *in-line* ассемблерного кода .

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: AsmLab4(C++).cpp

```
#include <iostream>
#include <fstream>

using namespace std;

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

int main() {
    system("chcp 1251 > nul");
    setlocale(LC_CTYPE, "rus");

    cout << "Азаров Максим 0382, формирование выходной строки
только из цифр и русских букв входной строки:" << endl;

    ofstream file;
    file.open("res.txt");

    cin.getline(input, 81);

    __asm {
        mov esi, offset input
        mov edi, offset output

        check_symb1 :
            lodsb

            cmp al, '\0'
            je finish

            cmp al, '0'
            jb check_symb1

            cmp al, '9'
            jbe write_symb1

            cmp al, 'ё'
            je write_symb1

            cmp al, 'Ё'
            je write_symb1

            cmp al, 'А'
            jb check_symb1

        write_symb1 :
            stosb
            jmp check_symb1
    }
```

```
        finish :  
            stosb  
    };  
  
    cout << output;  
    file << output;  
    file.close();  
    return 0;  
}
```