МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра вычислительной техники

**Лабораторная работа №5**

**по дисциплине «Технология программирования»**

|  |  |
| --- | --- |
| Студент | Ванин К.Е. |
| Группа | АВТ-819 |
| Преподаватель | Гаврилов К. В. |

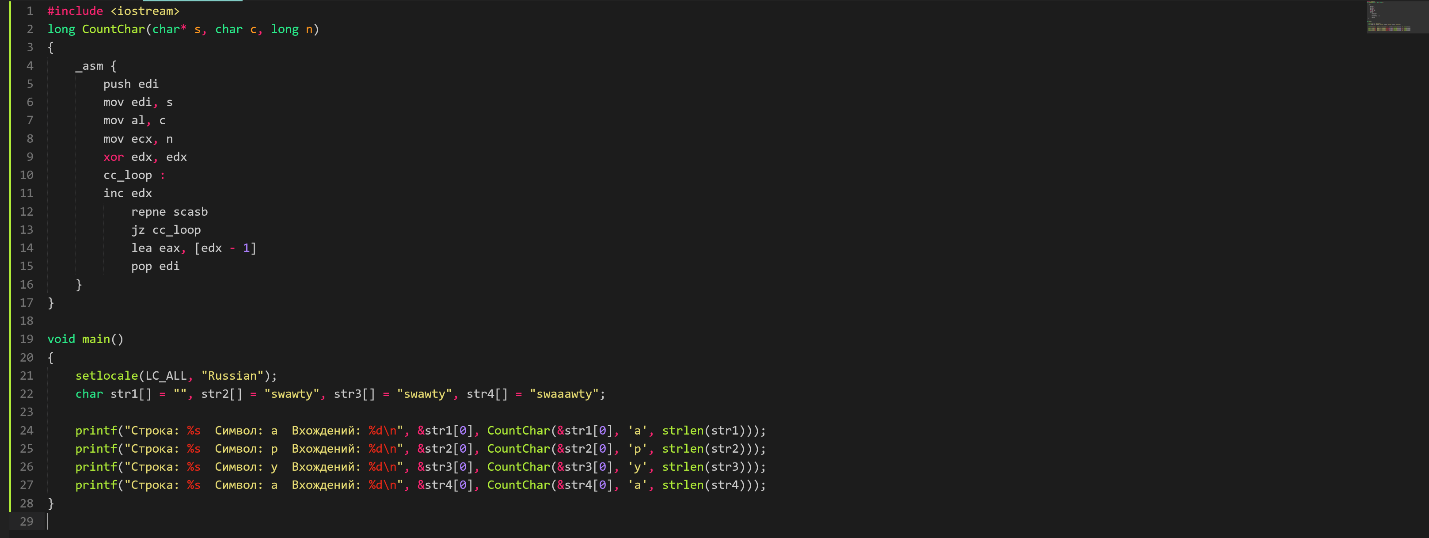
Новосибирск

2020

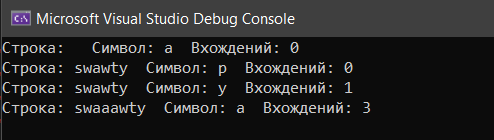
**Задание.**

1. Ввести с клавиатуры строку символов в переменную типа string и символ — в переменную типа char.
2. Вызвать функцию CountChar, которая написана в виде ассемблерной вставки. В качестве параметров в функцию передаются адрес начала строки, символ и длина строки.
3. Найденное количество вхождений символа вывести на экран.

**Листинг:**

****

**Результат:**

****

**Работа алгоритма:**

1. Регистр **edi** может хранить в себе нужную информацию. Значение **edi** заносится в машинный стек, чтобы вернуть его на место после выполнения целевой задачи задачи.
2. Адрес первого символа строки заносится в регистр **edi**
3. Заносится символ, который будем искать в младший байт **eax** – **al**.
4. Заносится длина строки в регистр **ecx**.
5. Зануляем все биты регистра **edx**
6. С помощью инкремента увеличиваем значение регистра **edx** на 1. Флаг при этом **ZF** = 0 , поскольку **edx** != 0
7. С помощью префикса **repne** идёт циклическое выполнение **scasb**
8. Командой **scasb** вычитаем из **al** (нужный символ) содержимое по адресу текущего элемента массива, при этом содержимое регистра и ячейки памяти не изменяется.

После этого значение регистра **edi** увеличивается на 1 (т.е. будет адрес следующего элемента массива). Значение увеличивается, поскольку флаг **DF = 0** по умолчанию.   
Если искомый символ равен символу в текущем элементе массива, то флаг **ZF** = 1 , иначе **ZF = 0**

1. Если **ecx** != 0, то выполняется команда **scasb** (пункт 7a), иначе – переходим к следующей команде (пункт 8).
2. Уменьшаем счетчик **ecx** на 1.
3. Если **ZF = 1** , то выполняем следующую команду (пункт 8), иначе переходим к пункту 7b.
4. Если флаг **ZF = 1**, то переходим по метке к 6 пункту, иначе – выполняется след. команда.
5. Из значения регистра **edx** вычитаем единицу и заносим данное значение в **eax**. В самом начале, в пункте 6, до проверки символа, добавлена лишняя единица, поэтому единицу нужно вычесть
6. Извлекается из стека и помещается обратно значение **edi**, которое сохранено в начале программы.

**Частные случаи:**

1. **Пустая строка**. -> в пункте 6, значение **edx** устанавливается равное 1.

В 7 пункте, поскольку длина строки **ecx** = 0, то переходим в 8 пункту.

В 8 пункте, поскольку в 6 пункте был установлен флаг **ZF** = 0, то переходим к 9 пункту.

Вычитаем из результата лишнюю 1 добавленную в 6 пункте.

1. **Нужный символ в строке не найден** -> 7 пункт повторяется **n** раз, при этом **ZF** = 0 после каждого выполнения **scasb**.

Цикл заканчивается, когда **ecx** = 0, переходим к 8 пункту. Поскольку **ZF** = 0 => переходим к 9 пункту и вычитаем лишнюю 1, добавленную в начале.

1. **Нужный символ в строке только один и при этом является последним символом**. -> В 7 пункте первые **n-1** итераций, устанавливают флаг **ZF = 0**, **n**-ая итерация устанавливает **ZF** = 1 и ecx=0. Т.к. zf=1, то переходим к 8, а потом к 6 пункту и увеличиваем значение регистра **edx**, при этом сбрасывается флаг **ZF** = 0.

Поскольку **ecx** = 0, то переходим к 8 пункту.

В 8 пункте, так как **ZF** = 0, то переходим к 9 пункту и вычитаем лишнюю 1.

1. **Нужный символ встречается в строке больше одного раза**. -> В очередной раз, когда мы нашли искомый символ в 7 пункте, то мы переходим к 6 пункту, увеличиваем значение **edx** и продолжаем циклическое выполнение команды **scasb**, пока не сравним все символы строки.