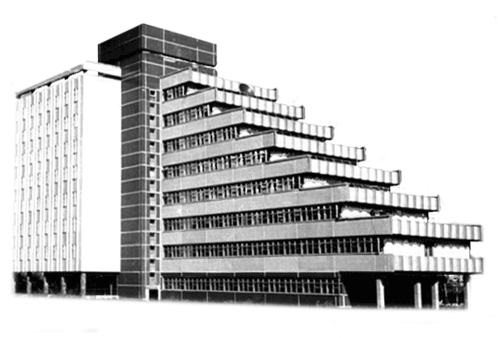
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное агентство по образованию

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Учебный Центр Информационных Технологий «Информатика»



Лабораторная работа № 4

по дисциплине «Информатика и программирование (I часть)»

Направление подготовки: 230105 - «Программное обеспечение вычислительной техники

и автоматизированных систем»

Выполнил слушатель: Коренский А.А.

Группа: 1

Вариант: 2

Дата сдачи: .03.2015

Преподаватель: Прытков Д.В.

Новосибирск, 2015г.

# Задание

В строке, содержащей последовательность слов, найти конец предложения, обозначенный символом "точка". В следующем слове первую строчную букву заменить на прописную.

# Теоретический материал

Базовый тип данных*char* понимается двояко: как байт -минимальная адресуемая единица представления данных в компьютере, и как символ текста. Это возможно по той причине, что в архитектуре компьютера (клавиатуре, экране, принтере, обслуживающих их программах -драйверах) в соответствии с общепринятым стандартом (см. Приложение 7) установлено соответствие между символами и присвоенными им значениями целой переменной (кодами). Любое устройство, отображающее символьные данные, при получении любого кода выводит соответствующий ему символ. Аналогично клавиатура (совместно с драйвером) кодирует нажатие любой клавиши с учетом регистровых и управляющих клавиш в соответствующий ей код.

Таким образом, символы преобразуются в цифровую форму, доступную для обработки в компьютере. Тип данных *char* не имеет никаких ограничений на выполнение операций, допустимых для целых переменных: от операций сравнения и присваивания до арифметических операций и операций с отдельными битами. Но за исключением редких случаев знание кодов символов не требуется. Для представления отдельных символов можно пользоваться символьными (литерными) константами, которые представляют собой символ, заключенный в одинарные кавычки (апострофы). Транслятор вместо такой константы всегда подставляет код соответствующего символа, например:

Имеется ряд кодов так называемых неотображаемых символов, которым соответствуют определенные действия при вводе-выводе символьных данных. Например, символу с кодом 0x0D (“возврат каретки”) соответствует перевод курсора в начало строки. Для их представления в программе используются символьные константы, начинающиеся с обратной косой черты.

Некоторые программы и стандартные функции обработки символов и строк (isdigit, isalpha) используют тот факт, что цифры, прописные и строчные (маленькие и большие) латинские буквы имеют упорядоченные по возрастанию значения кодов.

Строкой называется последовательность символов, ограниченная символом с кодом 0, то есть '\0'. Из ее определения видно, что она является объектом переменной размерности. Поскольку в Си допускаются переменные только фиксированной размерности, то единственным допустимым способом представления строки является массив символов. Здесь сразу проясним суть взаимоотношений между строкой и массивом символов:

-строка хранится в массиве символов, массив символов может быть инициализирован строкой, а может быть заполнен программно.

-строка представляет собой последовательность, ограниченную символом '\0', поэтому работать с ней можно, просматривая ее от начала до конца, пока не встретится символ '\0':

-соответствие размерности массива и длины строки транслятором не контролируется, за это несет ответственность программа (программист, ее написавший.

Строковой константой называется последовательность символов, заключенная в двойные кавычки. Допустимо использование неотображаемых символов. Строковая константа автоматически дополняется символом '\0', ею можно инициализироваться массив, в том числе такой, размерность которого определяется размерностью строки.

# Описание алгоритма

Цель данной программы – в строке найти конец предложения, обозначенный символом "точка" и в следующем слове первую строчную букву заменить на прописную. Алгоритм нахождения искомого значения представлен ниже.

Достижение заданной цели реализовано с помощью функции FourLab(). Ее можно условно разбить на три пункта, состоящих из трех циклов *for*. Первый цикл задает перебор символов строки, пока не встретит символ точки. Второй цикл задает перебор символов после точки. Третий выполняет основную работу, проверяя *i-й* элемент массива на равенство прописным, строчным латинским буквам, а также на условие начала слова (предыдущий символ равен ‘ ’). При истинности этой проверки *i-*й элемент массива заменяется на свой прописной аналог. После, при помощи функции *goto(),* выполнение переходит к следующей итерации первого цикла *for* и обрабатывается оставшаяся часть строки.

Printf(“Sum”);

Sum+=M[h];

Rmin+=abs(M[h]);

For

(h=0;h<N;h++)

# 4. Описание реализации

Подключаемые библиотеки:

#include "stdio.h"

#include "conio.h"

#include "stdlib.h"

#include "math.h"

Используемые функции:

int main() – вызывает функцию *FourLab()*;

printf() – вывод форматированного текста на экран;

goto() – для перехода к строке кода с меткой (осуществляет выход сразу из трех циклов);

getch() – ожидание ввода символа с клавитуры.

Используемые конструкции:

if – условный оператор, для создания ветвления ;

for – оператор цикла;

#define – препроцессорная директива (подстановка значения по всему коду программы).

# 5. Пример работы программы

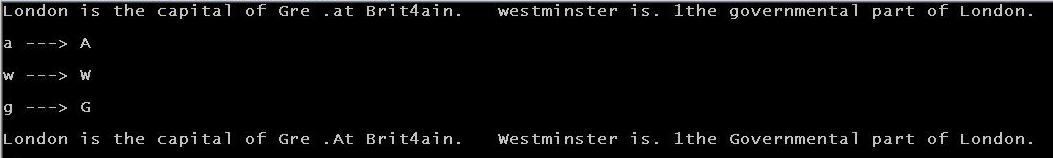


Рисунок 1. Результат работы функции *FourLab()*

На рисунке 1 показан результат работы функции *FourLab ()*. Вначале она выводит на экран заданный массив, затем поочередно строчные буквы, преобразовавшиеся в прописные. По окончании перебора всех элементов, на экран выводится массив с новыми значениями.

**Выводы**

Перед выполнением этой работы, был изучен теоретический материал по данным темам. Затем необходимо было продумать алгоритм будущей программы, для общего представления ее конструкции. После чего, я приступил к написанию исходного кода.

В ходе выполнения лабораторной работы были изучены принципы со строками: ввод/вывод/обработка.

После было проведено контрольное тестирование программы, которое показало правильность ее работы.

**Приложение. Текст программы с комментариями**

#include "stdio.h"

#include "conio.h"

#include "math.h"

#include "stdlib.h"

#define N 500

//ЛАБАРТОРНАЯ РАБОТА №4 ВАРИАНТ 2

//ЗАДАНИЕ: В СТРОКЕ, СОДЕРЖАЩЕЙ ПОСЛЕДОВАТЕЛЬНОСТЬ СЛОВ, //НАЙТИ КОНЕЦ ПРЕДЛОЖЕНИЯ, ОБОЗНАЧЕННЫЙ СИМВОЛОМ //"ТОЧКА". В СЛЕДУЩЕМ СЛОВЕ ПЕРВУЮ СТРОЧНУЮ БУКВУ

//ЗАМЕНИТЬ НА ПРОПИСНУЮ (ЗАГЛАВНУЮ)

int main() {

char FourLab(); //объявление функции

FourLab(); //вызов функции

getch();

}

int FourLab() {

int i, k, n;

char A[N]="London is the capital of Gre .at Brit4ain. westminster is. 1the governmental part of London.\n";

printf("%s\n", A);

for (i=0;i<N;i++) { //перебор символов, пока не найдет символ точки

if (A[i]==46) {

i++; //переход к следующему за точкой символу

for(;i<N;i++) { //задает перебор символов после точки

for(k=97,n=0;k<123;k++){ //проверяет i элемент на

//равенство строчным

if ((A[i]==k || A[i]==(65+n)) && ((A[i-1]==' ') || (A[i-1]=='.'))) {

printf("%c ---> ", A[i]);

A[i]=65+n; //заменяет на заглавную

//букву (A = 65 (ASCII))

printf("%c\n\n", A[i]);

goto ZZ;

}

n++;

}

}

}

ZZ:;

}

printf("%s", A);

}

Защита:

1)

void F3(char c[])

{ int i;

for (i=0; c[i] !='\0'; i++)

if (c[i] >='a' && c[i] <='z')

c[i] += 'A' - 'a';

}

Преобразует строчные латинские буквы в прописные.

2)

void F5(char c[])

{ int i,j;

for (i=0, j=0; c[i] !='\0'; i++)

if (c[i] !=' ') c[j++] = c[i];

c[j] = '\0';

}

Удаляет из строки все пробелы.