*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана***  ***(национальный исследовательский университет)»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ \_\_\_\_\_ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА \_\_\_\_\_\_\_ Компьютерные Системы и сети (ИУ6)\_\_\_\_\_\_\_\_\_\_\_

**Отчет**

**по домашней работе № 1**

**Вариант 22**

**Дисциплина: машинно-зависимые языки и основы компиляции**

**Название лабораторной работы:** **обработка символьной информации**

Студент гр. ИУ6-44  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Шумаков А.А.**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Данилюк С.С.**

(Подпись, дата) (И.О. Фамилия)

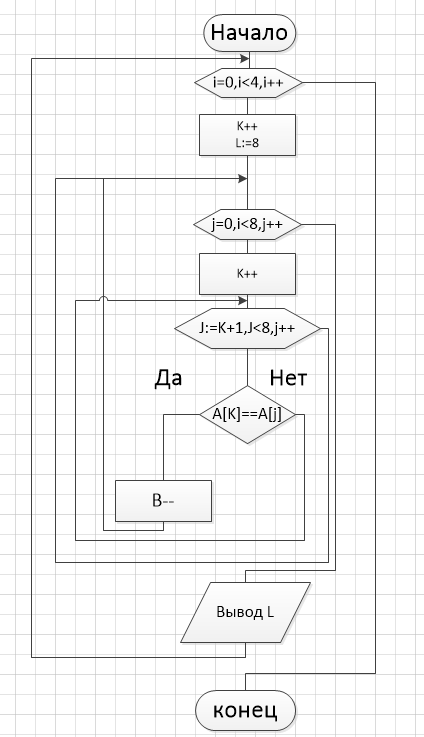
Москва, 2018

**Задание**

Дан текст, состоящий из 4 слов по 8 символов. Определить количество различных букв в каждом слове.

**Выполнение**

Прежде всего на рисунке 1 приведем схему алгоритма данной программы.



**Рис.1 – Схема алгоритм**

**Код программы**

*; Template for console application*

*.586*

*.MODEL flat, stdcall*

*OPTION CASEMAP:NONE*

*Include kernel32.inc*

*Include masm32.inc*

*IncludeLib kernel32.lib*

*IncludeLib masm32.lib*

*.CONST*

*MsgExit DB 13,10,"Press Enter to Exit",0AH,0DH,0*

*Enter\_Msg DB "Your string",0AH,0DH,0*

*mystr DB "aveaveme ggwwppxe yesbesit kekdzlol ",0 ; строка данная по условию*

*res DB 0AH,0DH,'Chislo razlichnix slov ',0AH,0DH,0*

*.DATA*

*Perenos DB 13,10*

*.DATA?*

*inbuf DB 100 DUP (?)*

*rezstr db 0AH,0DH,22 DUP (' '),0AH,0DH,0*

*kol sword ?*

*letters sword ? ; Количество различных букв в слове*

*help sdword ? ; Переменная для перехода к оставшимся в слове буквам, смещение по буквам в слове*

*.CODE*

*Start:*

*Invoke StdOut,ADDR Enter\_Msg*

*Invoke StdOut,ADDR mystr*

*Invoke StdOut,ADDR Perenos ; Для лучшего вывода*

*lea EDI,[mystr-1] ; в EDI работа со строкой*

*lea ESI,[mystr-1] ; в ESI хранится строка*

*mov EBX,4 ; количество слов в строке*

*cld ; сброс флага DF-обработка от начала к концу строки*

*cycle1: cmp BX,0 ; конец слов?*

*je conec ; все слова обработали, переход на конец*

*dec BX ; уменьшаем количество слов*

*mov DX,8 ; количество букв в слове в DX*

*mov letters,8 ; в переменную letter количество различных букв в слове, полагаем что они все разные*

*; если не разные то будем вычитать из letters*

*mov help,0 ; начальное смещение по буквам в слове*

*add edi,1 ; сдвигаем строку на символ вперед*

*add esi,1 ; сдвигаем строку на символ вперед*

*cycle2:*

*cmp DX,0 ; остались ли буквы в слове?*

*je vivod ; переход по viv, если не осталось букв в слове*

*mov EDI,ESI ; восстанавливаем в EDI строку для пробега по оставшимся буквам*

*dec DX ; уменьшаем количество оставшихся букв в слове*

*lodsb ; извлекаем из ESI символ и заносим в регистр A0L для дальнейшего поиска*

*mov ECX,7 ; в ECX количество оставшихся букв*

*sub ECX,help ; Вычитаем из CX "пройденные буквы"*

*add EDI,1 ; смещаем EDI на 1 символ*

*cycle: cmp ECX,0 ; остались ли буквы после вычитания "пройденных букв"*

*je cycle3 ; если букв не осталось переход по cycle3*

*repne scasb ; ищем символ в EDI=AL, если не нашли то в ECX 0*

*je equal ; символы одинаковы*

*jmp cycle ; если нет символов то переход в начало, нашли*

*equal: dec letters ; уменьшаем количество различных букв в текущем слове*

*cycle3: add help,1 ; увеличиваем смещение букв*

*jmp cycle2*

*; выводим количество разных букв в слове*

*vivod: push ECX*

*Invoke StdOut,ADDR res*

*Invoke dwtoa,letters,ADDR rezstr*

*Invoke StdOut,ADDR rezstr*

*pop ECX*

*jmp cycle1*

*conec: XOR EAX,EAX*

*Invoke StdOut,ADDR MsgExit*

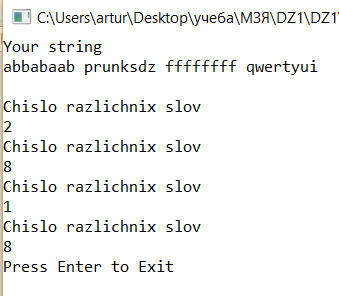
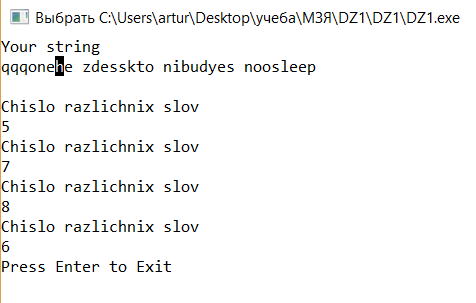
*Invoke StdIn,ADDR inbuf,LengthOf inbuf*

*Invoke ExitProcess,0*

*End Start*

**Тесты программы**

|  |  |  |
| --- | --- | --- |
| Исходные данные | Ожидаемый результат | Полученный результат |
| aveaveme ggwwppxe yesbesit kekdzlol | 4 5 6 6 | 4 5 6 6 |
| abbabaab prunksdz ffffffff qwertyui | 2 8 1 8 | 2 8 1 8 |
| qqqonehe zdesskto nibudyes noosleep | 5 7 8 6 | 5 7 8 6 |
| kokococo colacoka coalacok calllock | 3 5 5 5 | 3 5 5 5 |

**Рис.2 – Работа программы**

**Вывод**

1. Из-за слишком большого и сложного контроля количества введенных символов, намного удобнее ввод слов организовать программно.
2. При поиске уникальных букв нельзя забывать про следующий за словом пробел, учитывать его при переходе к следующему слову.
3. Программу можно легко переделать под поиск букв в словах любой длинны. Достаточно изменить одну переменную.