

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ «БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»
(БГТУ им. В.Г.Шухова)**

Лабораторная работа №7
дисциплина «Архитектура вычислительных систем»
по теме «Цепочечные команды»

Выполнил: студент группы ВТ-31
Проверил:

Макаров Д.С.
Осипов О.В.

Белгород 2019

Лабораторная работа №7

«Цепочечные команды»

Цель работы: изучение цепочечных команд для работы с одномерными массивами и строками.

Задание:

Написать программу для решения соответствующего варианта задания. По умолчанию, если в задании не оговорено, считать что в тексте могут быть только буквы русского алфавита, латинские буквы, цифры, пробелы и нулевой символ как признак окончания строки. Пробелов между словами может быть несколько. Необходимые операции копирования, сравнения, поиска и другие организовать в виде подпрограмм, используя цепочечные команды. Если в результате преобразования длина строки увеличивается, то изначально зарезервировать для неё большой объём памяти.

Ход работы

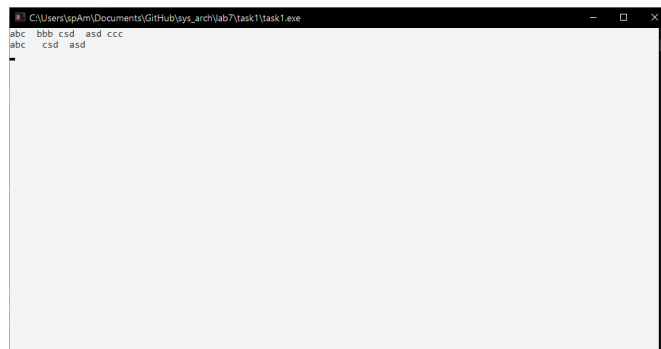


Рис. 1: Пример работы программы

Приложение

Содержимое файла task1.asm

```
.386
.model flat, stdcall
option casemap: none

include c:\masm32\include\windows.inc
include c:\masm32\include\kernel32.inc
include c:\masm32\include\user32.inc
include c:\masm32\include\msvcrt.inc
includelib c:\masm32\lib\user32.lib
includelib c:\masm32\lib\kernel32.lib
includelib c:\masm32\lib\msvcrt.lib

.DATA
a db "abc  bbb csd  asd ccc",0
format_str db "%s", 10,13,0
.CODE

;1 аргумент - указатель на строку
;2 аргумент - индекс для удаления
delete_char proc
    push eax
    push ebx
    push ecx
    push edx

    xor eax,eax
    xor ebx,ebx
    xor ecx,ecx
    xor edx,edx

    mov eax,[esp+24]
    mov ebx,[esp+20]
    mov ecx, ebx
    add ecx,1
    dec ebx
    dec ecx
loop_start:
    inc ebx
    inc ecx
    mov dl,byte ptr[eax+ecx]
    mov byte ptr[eax+ebx],dl
    mov dl,byte ptr[eax+ebx]
    cmp edx,0
    jne loop_start
    pop edx
    pop ecx
    pop ebx
    pop eax
    ret 12
delete_char endp

START:
PUSH offset a
PUSH offset format_str
call crt_printf
ADD ESP, 8
```

```

mov ebx,offset a - 1
work_cycle:
    find_cycle:
        ;ebx - текущий указатель
        add ebx,1
        mov ecx,[ebx]
        cmp cl,0
        je work_end
        mov ecx,[ebx]
        cmp cl,20h
        jle find_cycle
    ;ebx - указатель на начало слова
    ;edx - рабочий указатель внутри слова
    mov edx,ebx
    mov eax,0
    check_cycle:
        add edx,1
        mov ecx,[edx]
        cmp cl,0
        je del_check
        mov ecx,[edx]
        cmp cl,20h
        jle del_check
        mov ecx,[edx]
        push eax
        mov eax,[ebx]
        cmp al,cl
        pop eax
        je check_cycle
        mov eax,1
        jmp check_cycle
    del_check_false:
        mov ebx,edx
        jmp find_cycle
    del_check:
        cmp eax,1
        je del_check_false
    del_cycle:
        push ebx
        push 0
        call delete_char
        mov ecx,[ebx]
        cmp cl,20h
        jnle del_cycle
        jmp find_cycle
work_end:

PUSH offset a
PUSH offset format_str
call crt_printf
ADD ESP, 8

END START

```