

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Департамент программной инженерии

**ПРОГРАММА, ЗАМЕНЯЮЩАЯ ВСЕ ГЛАСНЫЕ БУКВЫ
В ЗАДАННОЙ ASCII-СТРОКЕ ЗАГЛАВНЫМИ**

Пояснительная записка

Исполнитель
Студент группы БПИ 196
Судаков Дмитрий Юрьевич

Москва, 2020

Содержание

1	Постановка задачи	2
2	Описание алгоритма	3
2.1	Описание методов	3
2.1.1	strLen	3
2.1.2	findInStr	3
2.1.3	transformString	3
2.2	Формат входных данных	3
2.3	Формат выходных данных	3
3	Приложение 1. Код программы	4
4	Приложение 2. Тестирование программы	6

1 Постановка задачи

Разработать программу, заменяющую все гласные буквы в заданной ASCII-строке заглавными.

2 Описание алгоритма

Для решения поставленной задачи было заведено два массива со всеми строчными и прописными гласными английскими и русскими буквами в кодировке OEM-866 (Такая кодировка была выбрана, так как она установлена по умолчанию в консоли). Далее осуществлялся поиск каждого символа из исходной строки в массиве строчных букв, и если поиск был успешным, то он заменяется на элемент с таким же индексом из массива прописных букв.

2.1 Описание методов

2.1.1 strLen

Аргументы : string ссылка на строку, заканчивающуюся символом '\0' длину которой надо посчитать.

Возвращаемое значение : В регистр eax помещается длина строки с учётом символа

2.1.2 findInStr

Аргументы : string Ссылка на строку в которой будет поиск
len0 Длина строки с учётом нуля в конце
char Символ, который надо найти

Возвращаемое значение : В регистр eax помещается -1, если символ не найден, или его индекс в строке, если найден

2.1.3 transformString

Описание : Метод заменят все гласные буквы в строке на их заглавные эквиваленты.

Аргументы : string Ссылка на строку в которой надо заменить символы

Возвращаемое значение : Изменяется переданная строка

2.2 Формат входных данных

Вводится строка состоящая из символов в кодировке OEM-866 длиной до 256 символов. Все символы, которые будут напечатаны сверх заданного лимита будут проигнорированы.

2.3 Формат выходных данных

Выводится сообщение с считанной строкой до и после преобразования.

3 Приложение 1. Код программы

```
;Develop a program that replaces all  
;vowels in a given ASCII string  
;in capital
```

```
format          PE console 4.0  
entry           start
```

```
include         'win32a.inc'
```

```
section         '.code' code readable executable
```

```
start:
```

```
    cinvoke printf, outEnterMessage      ;print message for user  
    cinvoke scanf, inStringTemp, myStr   ;read users string  
    cinvoke printf, endl                 ;go to next line
```

```
    cinvoke printf, outStringBefore, myStr ;print users line
```

```
    stdcall transformString, myStr        ;tranform string
```

```
    cinvoke printf, outStringAfter, myStr ;print transformed string
```

```
exit:
```

```
    cinvoke _getch
```

```
    invoke ExitProcess, 0
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
;Function compute length of string, uncluding zero-symbol in the end
```

```
;Returns result in eax
```

```
proc strlen uses ecx edi, string
```

```
    xor al, al          ;clear al for comparing with string elems
```

```
    mov ecx, -1          ;ecx = -1 because if rep-cycle decrement ecx and stopped when ecx = 0
```

```
    mov edi, [string]    ;set edi on first element of our string
```

```
    cld                  ;set DF for scasb command
```

```
    repnz scasb          ;while ([edi] != al) {edi++; ecx--}
```

```
                        ;now ecx = -1 -(len+1) = -len-2
```

```
    not ecx              ;ecx = len+1
```

```
    mov eax, ecx         ;return result with eax
```

```
    ret
```

```
endp
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
;Fuction try to find number of given char in string
```

```
;Returns in eax number of char, or -1 if it hasn't found
```

```
proc findInStr uses ebx ecx edi, string, len0, char
```

```
    mov ecx, [len0]      ;ecx = len + 1
```

```

    mov ebx, [len0]          ;ebx = len + 1
    dec ebx                  ;ebx = len

    mov al, BYTE[char]       ;set to al our char
    mov edi, [string]        ;set edi on first element of our string
    cld                      ;set DF for scasb command

    repnz scasb              ;while ([edi] != al) {edi++; ecx--}

                                ;now in ecx number of char from the end
    test ecx, ecx            ;compare ecx with 0
    mov eax, -1              ;if it equal to 0, set eax to -1
    jz @f                    ;jump to the end of proc

    mov eax, ebx              ;eax = len
    sub eax, ecx              ;eax = number of our char
@@:
    ret
endp

;////////////////////////////////////

;Method transformed all lowercase vowels to uppercase
proc transformString uses eax ecx esi, string

    stdcall strlen, [string] ;eax = len + 1
    mov ecx, eax             ;ecx = len + 1
    dec ecx                  ;ecx = len

    test ecx, ecx            ;if string empty jump to the end
    je .exit

    mov esi, [string]        ;set to esi adress of given string
    cld                      ;set DF for lodsb command

.forLoop:
    xor eax, eax              ;clear eax
    lodsb                    ;move next symbol to al
    stdcall findInStr, fromString, [fromStringLen0], eax
                                ;find this symbol in fromString
    cmp eax, -1              ;if eax == -1 go to the next symbol
    je @f

    dec esi                  ;back to prev symbol
    mov al, BYTE[toString+eax] ;move to al capital leter
    mov [esi], al            ;move to [esi] our letter
    inc esi                  ;move esi to the next symbol

@@:
    loop .forLoop

.exit:
    ret
endp

;////////////////////////////////////

section          '.data' data readable writeable

```


Попробуем русский текст :

```
S:\SoftCraft\sources\CONSOLE\MicroProject_1>program.EXE
Enter Your string (no more than 255 symbols): Товарищи! сложившаяся структура организации представляет собой интересный эксперимент проверки на
правлений прогрессивного развития.

String before : Товарищи! сложившаяся структура организации представляет собой интересный эксперимент проверки направлений прогрессивного разви
тия.
String after : ТовАрищи! слоЖившАяся стрУктУрА ОргАнизАции прЕдстАвляет сОбой ИнтЕрЕсный ЭкспЕримЕнт прОвЕрки нАпрАвлЕний прОгрЕссивнОгО рАзвИт
иЯ.
```

Попробуем английский текст :

```
S:\SoftCraft\sources\CONSOLE\MicroProject_1>program.EXE
Enter Your string (no more than 255 symbols): Ius dicat feugiat no, vix cu modo dicat principes. Mandamus abhorreant deseruisse mea at, mea eli
t deserunt persequeris at, in putant fuisset honestatis qui.

String before : Ius dicat feugiat no, vix cu modo dicat principes. Mandamus abhorreant deseruisse mea at, mea elit deserunt persequeris at, in
putant fuisset honestatis qui.
String after : IUс dIcAt fEUGIAt nO, vIx cU mOdO dIcAt prIncIpEs. MAndAmUs AbhOrrEAnt dEsErUIssE mEA At, mEA ElIt dEsErUnt pErSEquErIs At, In p
UtAnt fUIssEt hOnEstAtIs qUI.
```

Попробуем комбинированный текст :

```
S:\SoftCraft\sources\CONSOLE\MicroProject_1>program.EXE
Enter Your string (no more than 255 symbols): Lorem ipsum dolor sit amet, an eos lorem ancillae expetenda, vim et utamur quaestio.Повседневная
практика показывает, что реализация намеченных плановых заданий в значительной степени обуславливает создание модели развития.

String before : Lorem ipsum dolor sit amet, an eos lorem ancillae expetenda, vim et utamur quaestio.Повседневная практика показывает, что реали
зация намеченных плановых заданий в значительной степени обуславливает создание модели развития.
String after : LOrEm IpsUm dOlOr sIt AmEt, An EOс lOrEm AncIllAE ExpEtEndA, vIm Et UtAmUr quAEstIO.ПовсЕднЕвнАЯ прАктикА пОкАзЫвАет, что РЕАлиЗ
АЦИЯ нАмЕчЕнных плАновЫх зАдАний в зНАчитЕльной стЕпени ОбУслАвлиВАет сОздАНИЕ мОдЕли рАзвИтия.
```

Попробуем цифры и другие символы :

```
S:\SoftCraft\sources\CONSOLE\MicroProject_1>program.EXE
Enter Your string (no more than 255 symbols): 1234567890{"}":?>?"?

String before : 1234567890{"}":?>?"?
String after : 1234567890{"}":?>?"?
```