

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Департамент программной инженерии

ПРОГРАММА ВЫЧИСЛЕНИЯ ЧИСЛА ГЛАСНЫХ И СОГЛАСНЫХ БУКВ В
ASCII-СТРОКЕ

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

Пояснительная записка
ЛИСТ УТВЕРЖДЕНИЯ
RU.17701729.04.01-01 81 01-1-ЛУ

Исполнитель
Студент группы БПИ192
_____/ Д.Е. Борисов/
«__» _____ 2020 г.

Москва 2020

УТВЕРЖДЕН
RU.17701729.04.01-01 81 01-1-ЛУ

**ПРОГРАММА ВЫЧИСЛЕНИЯ ЧИСЛА ГЛАСНЫХ И СОГЛАСНЫХ БУКВ В
ASCII-СТРОКЕ**

Пояснительная записка

RU.17701729.04.01-01 81 01-1-ЛУ

Листов 12

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Москва 2020

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ	3
1.1 Наименование программы	3
1.2 Текст задачи.....	3
2. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ.....	4
2.1 Описание алгоритма и функционирования программы	4
2.1.2 Переменные для промежуточных и исходных данных.....	4
2.1.3 Описание алгоритма работы программы.....	4
2.1.3.1 Чтение данных	4
2.1.3.2 Алгоритм подсчета гласных/согласных.....	4
2.1.3.3 Вывод данных	4
2.3 Организация входных данных	5
2.4 Организация выходных данных.....	5
3. Тестирование программы	6
4. СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ.....	7
ПРИЛОЖЕНИЕ1. ТЕКСТ ПРОГРАММЫ.....	8
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....	12

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 81				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

1. ВВЕДЕНИЕ

1.1 Наименование программы

Наименование программы – «программа вычисления числа гласных и согласных букв в ASCII-строке».

1.2 Текст задачи

Разработать программу вычисления числа гласных и согласных букв в ASCII-строке.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

RU.17701729.04.01-01 81 01-1
2. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

2.1 Описание алгоритма и функционирования программы

2.1.2 Переменные для промежуточных и исходных данных

- inputStr - расположение введенной пользователем строки.
- numberVowels – адрес для переменной подсчета количества гласных в строке.
- numberConsonants - адрес для переменной подсчета количества согласных в строке.
- Также в программе присутствует ряд переменных, служащих для ввода-вывода информации и подсказок пользователю.

2.1.3 Описание алгоритма работы программы

Программа разбита на 3 блока: чтение входных данных, подсчет количества согласных и гласных в строке, вывод подсчитанных значений в консоль.

2.1.3.1 Чтение данных

Чтение данных производится через вызов процедуры readInput.

Пользователю вводится приглашение для ввода строки: «Input your string: »б после чего с помощью вызова макроса scanf производится чтение введенной строки в переменную по адресу inputStr.

2.1.3.2 Алгоритм подсчета гласных/согласных

Сам алгоритм реализован в процедуре countAnswer. На вход процедуре в стеке подается строка для подсчета ответа в ней. Результат процедуры записывается в переменные numberVowels и numberConsonants – количество гласных и согласных в строке соответственно. На первом этапе алгоритма строка из стека переносится в регистр esi для использования цепочной функции lodsб и устанавливается флаг направления. После этого в цикле символ строки считывается в регистр еах посредством lodsб. Если считанный символ нулевой, то осуществляется выход из процедуры. В зависимости от того, попадает ли символ в промежутки ['А', 'Z'], ['а', 'z'] или не попадает программа переходит по меткам greaterChar, lowerChar или считывает новый символ. В метках greaterChar и lowerChar происходит сравнение считанной буквы с каждой из существующих гласных в нужном регистре (заглавном и строчном). В случае совпадения осуществляется переход на метку incVowel, в противном случае на метку incConsonants. В каждой из меток осуществляется инкремент значений по адресам numberVowels и numberConsonants соответственно. После чего процедура продолжает цикл.

2.1.3.3 Вывод данных

Вывод данных осуществляется в процедуре printAnswer. В ней последовательно печатаются строки «Number of vowels: <numberVowels>» и «Number of consonants: <numberConsonants>», каждая на новой строке.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

2.3 Организация входных данных

Программа считывает из консоли введенную пользователем строку в кодировке ASCII. Конец строки определяется пробелом или переходом на новую строку. Размер строки не должен превышать 1000 символов.

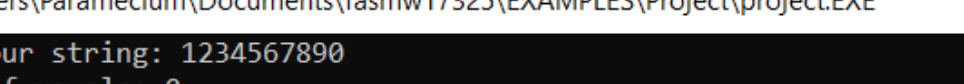
2.4 Организация выходных данных

В результате выполнения программы в консоль выводятся два сообщения, описанные в пункте 3.1.3.3. Первая строка содержит информацию о количестве гласных, вторая о количестве согласных.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

RU.17701729.04.01-01 81 01-1
3. Тестирование программы

Для подтверждения работоспособности программы ниже приведены результаты работы на некоторых входных данных.

[illegible]

The screenshot shows a Windows command prompt window with a title bar that reads "C:\Users\Paramecium\Documents\fasmw17325\EXAMPLES\Project\project.EXE". The command prompt has a black background and white text. The user has entered the string "1234567890" in response to the prompt "Input your string:". The program has calculated and displayed "Number of vowels: 0" and "Number of consonants: 0". The prompt "Press any key to exit..." is shown at the bottom, with a small white cursor character visible.

```
C:\Users\Paramecium\Documents\fasmw17325\EXAMPLES\Project\project.EXE
Input your string: 1234567890
Number of vowels: 0
Number of consonants: 0
Press any key to exit... █
```

```
C:\Users\Paramecium\Documents\fasmw17325\EXAMPLES\Project\project.EXE
Input your string: aabbcc
Number of vowels: 2
Number of consonants: 4
Press any key to exit...
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

4. СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

- 1) Документация FASM [Электронный ресурс] / Режим доступа
<http://flatassembler.narod.ru/fasm.htm>, свободный. (дата обращения: 28.10.2020)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 1. ТЕКСТ ПРОГРАММЫ

```

;-----Author: Borisov Dmitry cbse192-----
;-----Variant: 4-----
;-----Assigment: count number of vowels-----
;-----and consonants in the input string-----
format PE console

```

entry start

include 'win32a.inc'

section '.data' data readable writable

endStr db 'Press any key to exit... ', 0;prints in the end of program

stringForInput db 'Input your string: ', 0;prints before user inputs string

endl db 10, 13, 0;end of line

numberStr db '%d', 0

stringStr db '%s', 0

inputStr db 1001 dup (?), 0;User input

numberVowelsStr db 'Number of vowels: %d', 0;Message for number of vowels in string

numbrerConsonantsStr db 'Number of consonants: %d', 0;Message for number of

consonants in string

numberVowels dd 0;mark for number of vowels

numberConsonants dd 0;mark for number of consonants

NULL = 0

section '.code' code readable executable

```

;-----Entry point of a programm-----
;-----

```

start:

call readInput

stdcall countAnswer, inputStr;Main program logic

add esp, 4

call printAnswer

call endProgrammText;Call getch and after close down program

push NULL

call [ExitProcess]

```

;-----
; Input: string to process as stack argument

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

RU.17701729.04.01-01 81 01-1

; Procedure counts number of vowels in numberVowels

; and number of consonants in numberConsonants

countAnswer:

mov esi, [esp + 4];using input string as stack argument

cld;direction from begin to end

xor ecx, ecx;End loop condition char equals to 0 (ecx)

loopStr:

lodsb;load char

cmp cl, al;if zero goto end of loop

je endSrc

cmp al, 'A';Less than 'A' - not a letter

jl loopStr;Read next char

cmp al, 'Z';Less than 'Z' and greater than 'A' - greater letter

jle greaterChar;Precess greater letter

cmp al, 'z';Greater than 'z' - not a letter

jg loopStr;Read next char

cmp al, 'a';Less than 'z' and greater than 'a' - lower letter

jge lowerChar;Precess lower letter

jmp loopStr;Read next char

lowerChar:

;If any vowel increase number of vowels

cmp al, 'a'

je incVowel

cmp al, 'e'

je incVowel

cmp al, 'i'

je incVowel

cmp al, 'o'

je incVowel

cmp al, 'u'

je incVowel

cmp al, 'y'

je incVowel

;If not increase number of consonants

jmp incConsonants

greaterChar:

;If any vowel increase number of vowels

cmp al, 'A'

je incVowel

cmp al, 'E'

je incVowel

cmp al, 'I'

je incVowel

cmp al, 'O'

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

```

        je incVowel
        cmp al, 'U'
        je incVowel
        cmp al, 'Y'
        je incVowel

        ;If not increase number of consonants
        jmp incConsonants
incVowel:
        inc [numberVowels];Increase number of vowels
        jmp loopStr;Read next char
incConsonants:
        inc [numberConsonants];Increase number of consonants
        jmp loopStr;Read next char
endSrc:
        ret;Return from procedure

;-----
; Procedure prints two strings with number of vowels in first
; and number of consonants in second
printAnswer:
        cinvoke printf, numberVowelsStr, [numberVowels];print number of vowels
        cinvoke printf, endl;print new string
        cinvoke printf, numbrerConsonantsStr, [numberConsonants];print number of
consonants
        cinvoke printf, endl;print new string
        ret

;-----
; Procedure prints text before the program will
; close down and calls getch
endProgrammText:
        cinvoke printf, endStr
        call [getch]
        ret

;-----
; Procedure prints prompt for enter string and reads string into
; strignForInput mark
readInput:
        cinvoke printf, stringForInput
        cinvoke scanf, stringStr, inputStr
        ret

```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

section '.idata' import data readable

```
library kernel, 'kernel32.dll', \
    msvcrt, 'msvcrt.dll'
```

```
import kernel, \
    ExitProcess, 'ExitProcess'
```

```
import msvcrt, \
    printf, 'printf', \
    scanf, 'scanf', \
    getch, '_getch'
```

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.04.01-01 ТЗ				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

[illegible]