

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

Факультет компьютерных наук
Образовательная программа бакалавриата «Программная инженерия»

**Микропроект №1 по дисциплине
«Архитектура вычислительных систем»**

Пояснительная записка

Исполнитель:
студент группы БПИ193
Царёв Кирилл

Москва 2020

Оглавление

1.	Текст задания	3
2.	Описание методов и алгоритмов	3
3.	Список используемых источников	3
4.	Тестовые примеры	3
5.	Текст программы.....	5

1. Текст задания

Разработать программу, находящую в заданной ASCII-строке первую слева на право последовательность N символов, каждый элемент которой определяется по условию "меньше предшествующего" (N=4)

2. Описание методов и алгоритмов

Считываем строку; Вычисляем её длину; Копируем её со сдвигом на один символ влево; Сравниваем посимвольно исходную и сдвинутую строки; Если условие выполняется, то увеличиваем текущую позицию сравнения и количество символов в подстроке; Если условие не выполняется, то увеличиваем текущую позицию сравнения и обнуляем количество символов в подстроке; Выполняем это до тех пор, пока количество символов в подстроке не будет равно четырем или не закончатся символы в строке; Если такая подстрока есть, то сначала пропускаем N символов, где N это текущая позиция сравнения, на которой мы вышли из цикла, минус 4; Затем копируем следующие 4 символа; Выводим подстроку; Если подстрока не нашлась, то выводим соответствующее сообщение.

3. Список используемых источников

- 1) <http://www.softcraft.ru/edu/comparch>
- 2) <http://flatassembler.narod.ru/fasm.htm>
- 3) <https://ravesli.com/assembler-stroki>

4. Тестовые примеры

```
Input a string:
йцукенгшщздорпавыфячсмить
Substring found:
чсми_
```

```
Input a string:
qwertyuioplkjhgfdsazxcvbnm
Substring found:
lkjh_
```

```
Input a string:
abcdefghijklmnopqrstuvwxyz
Substring not found
```

```
Input a string:  
zyxwvutsrq  
Substring found:  
yxwv_
```

```
Input a string:  
1234567890  
Substring not found
```

```
Input a string:  
543  
Substring not found
```

```
Input a string:  
5432  
Substring not found
```

```
Input a string:  
54321  
Substring found:  
4321_
```

```
Input a string:  
5555555555554321  
Substring found:  
4321
```

```
Input a string:  
5555432111198765555  
Substring found:  
4321
```

5. Текст программы

format PE console
entry start

```
include 'win32a.inc'
;-----
;   Tsaryov Kirill group 193
;
;   First substring with length 4 with condition
;   "each symbol is less than previous"
;-----
; Data section
section '.data' data readable writable
    msg    db  'Input a string: ', 10, 0
    strScan db  '%s', 0
    string1 rb  256 ; Input string
    string2 rb  256 ; Shifted string
    string3 rb  256 ; Output substring
    len    dd  0
    nfStr   db  'Substring not found', 10, 0
    fStr    db  'Substring found: ', 10, 0
;-----
; Main program
section '.code' code readable executable
start:
    invoke printf, msg
    ; Scanning a string
    invoke scanf, strScan, string1

    ; Finding length of the string
    stdcall strlen
    ; If length < 5, then there is no substring
    cmp    [len], 5
    jl     NotFound

    ; Copying string1 to string2 with left shift by 1
    cld
    sub    [len], 1
    mov    ecx, [len]
    mov    esi, string1
    ; Skip 1st symbol
    movsb
    mov    edi, string2
Loop1:
    movsb
    loop  Loop1

    ; Comparing strings 1 and 2 by symbols
    mov    ecx, [len]
    ; Current number of symbols in substring
    xor    eax, eax
    ; Current position of comparison
    xor    ebx, ebx
    mov    esi, string2
    mov    edi, string1
    cld
```

```

Loop2:
    inc    ebx
    ; Comparing symbols
    cmpsb
    ; If condition is true
    jl     Tr
    ; Else
    jmp     Fls
Tr:
    ; Increase current number of symbols in substring
    inc    eax
    jmp     EndL
Fls:
    ; Undo current number of symbols in substring
    xor    eax, eax
    jmp     EndL
EndL:
    cmp    eax, 4
    je     Found
    loop   Loop2
    jmp     NotFound

Found:
    invoke printf, fStr

    cmp    ebx, 4
    jne    Not4

    ; If the last position of comparison is 4
    ; Save the substring
    cld
    mov    esi, string2
    mov    edi, string3
    mov    ecx, ebx
Loop0:
    movsb
    loop   Loop0
    jmp     EndF

; Else
Not4:
    ; Skip symbols before substring
    cld
    mov    esi, string2
    sub    ebx, 4
    mov    ecx, ebx
Loop3:
    movsb
    loop   Loop3
    ; Save the substring
    mov    edi, string3
    mov    ecx, 4
Loop4:
    movsb
    loop   Loop4
    jmp     EndF

EndF:
    invoke printf, string3

```

```

    jmp    Fin

NotFound:
    invoke printf, nfStr
    jmp    Fin

Fin:
    invoke getch
    invoke ExitProcess, 0

;-----
; Strlen procedure (include zero symbol)
strlen:
    mov    edi, string1
    mov    ecx, -1    ; ecx must be < 0
    xor    al, al    ; tail symbol is zero
    cld          ; direction from begin to end
    repne scasb    ; while(string[i] != al) {i++; ecx--;}
    neg    ecx
    sub    ecx, 2    ; ecx = length(string)
    mov    [len], ecx
    ret

;-----
; Including External Api
section '.idata' import data readable
    library kernel, 'kernel32.dll',\
        msvcrt, 'msvcrt.dll'

include 'api\kernel32.inc'

import kernel,\
    ExitProcess, 'ExitProcess'
import msvcrt,\
    printf, 'printf',\
    scanf, 'scanf',\
    getch, '_getch'

```