МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

# Кафедра электронные вычислительные машины

Отчет по предмету

Программирование на языке ассемблера

Лабораторная работа №2

**«СОЗДАНИЕ ПРОСТОЙ ПРОГРАММЫ НА ЯЗЫКЕ АССЕМБЛЕРА »**

|  |  |
| --- | --- |
| **Выполнил:**  Студент группы 130501  Гнездилов А. М. | **Проверил:**  Туровец Н. О. |

Минск 2022

**Цель работы**: Ознакомиться с директивами определения данных, изучить команды пересылки данных и передачи управления, изучить строчные операции и прерывания консольного ввода-вывода высокого уровня. **Задание:** Написать программу, которая выводит на экран «Hello, world»

**Код программы:**

.model small

.stack 100h

.data

msg1 db "Input string:", 0dh, 0ah, '$'

msg2 db 0dh, 0ah, "Enter the substring you want to delete:", 0dh, 0ah, '$'

msg3 db 0dh, 0ah, "Result: $"

msg4 db 0dh, 0ah, "Enter new substring: $"

string db 202 dup("$")

sbstrToRemove db 202 dup("$")

sbstrToInsert db 202 dup("$")

capacity EQU 200

flag dw 0

.code

main proc

mov ax, @data

mov ds, ax

mov es, ax

mov ah, capacity

mov string[0], ah ;first byte - max srting size

mov sbstrToRemove[0], ah

mov sbstrToInsert[0], ah

lea dx, msg1

call puts

lea dx, string

call gets

lea dx, msg2

call puts

lea dx, sbstrToRemove

call gets

lea dx, msg4

call puts

lea dx, sbstrToInsert

call gets

xor cx, cx

mov cl, string[1]

sub cl, sbstrToRemove[1]

jb End

inc cl

cld

lea si, string[2]

lea di, sbstrToRemove[2]

call ReplaceSubstring

End:

lea dx, msg3

call puts

lea dx, string[2]

call puts

mov ah, 4ch

int 21h

ret

endp main

ReplaceSubstring proc

Cycle:

mov flag, 1

push si

push di

push cx

mov bx, si

xor cx, cx

mov cl, sbstrToRemove[1]

repe cmpsb

je FOUND

jne NOT\_FOUND

FOUND:

call DeleteSubstring

mov ax, bx

call InsertSubstring

mov dl, sbstrToInsert[1] ;

add string[1], dl ;

mov flag, dx

NOT\_FOUND:

pop cx

pop di

pop si

add si, flag

Loop Cycle

ret

endp ReplaceSubstring

DeleteSubstring proc

push si

push di

mov cl, string[1]

mov di, bx

repe movsb

pop di

pop si

ret

endp DeleteSubstring

InsertSubstring proc

lea cx, string[2] ; string 1st symbol address

add cl, string[1] ; add string length to get to next symbol after the last

mov si, cx ; last symbol as a source

dec si ; at the last symbol

mov bx, si ; save last symbol in bx

add bl, sbstrToInsert[1] ; now there is the last symbol of new string in bx

mov di, bx ; new last symbol is reciever

;inc bx

mov dx, ax ; ax is a place to insert

sub cx, dx ; after last symbol -= place to insert

std ; moving backward

repe movsb

lea si, sbstrToInsert[2] ; source is sbstr 1st symbol

mov di, ax ; reciever is a place to insert

xor cx, cx ; set cx to zero

mov cl, sbstrToInsert[1] ; sbstr length to cx

cld ; moving forward

repe movsb

ret

endp InsertSubstring

; I/O procedures

gets proc

mov ah, 0Ah

int 21h

ret

endp gets

puts proc

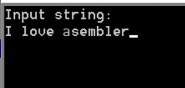
mov ah, 9

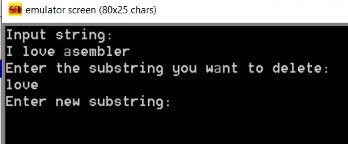
int 21h

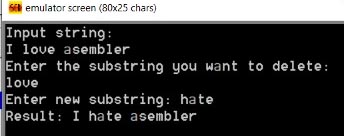
ret

endp puts

**Результат выполнения задания:**

****

****

****

**Вывод**

В ходе выполнения лабораторной работы ознакомился с директивами определения данных, изучил команды пересылки данных и передачи управления, изучил строчные операции и прерывания консольного ввода-вывода высокого уровня.