Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина «Конструирование программ»

**ОТЧЕТ**

к лабораторной работе №4

на тему:

**«ОБРАБОТКА СИМВОЛЬНЫХ ДАННЫХ»**

БГУИР 1-40-04-01

|  |
| --- |
| Выполнил студент группы 253504  Дмитрук Богдан Ярославович |
|  |
| (дата, подпись студента) |
| Проверил ассистент кафедры информатики  Романюк Максим Валерьевич |
|  |
| (дата, подпись преподавателя) |

Минск 2023

**Цель работы:** Вариант 7. Удалить заданное слово в строке.

**Ход работы:** Был создан файл .inc, реализующий следующие макросы: внесения в регистр ds адреса сегмента .data, вывода строки, посимвольного вывода строки, ввода строки, получения размера строки, получения символа строки по индексу, вывода символов новой строки, получения индекса первого вхождения подстроки в строку(в случае отсутствия подстроки в регистр ax вместо индекса первого вхождения заносится -1), удаление подстроки из строки, удаление слова из подстроки, помещение в стек набора регистров, изъятие из стека набора регистров, проверка наличия пробелов в строке.

Поиск индекса первого вхождения подстроки:

find\_substr macro bstr, fstr

get\_size str1

mov al, dl

get\_size str2

mov bl, dl

mov cl, al

xor ch, ch

xor bh, bh

xor di, di

xor ax, ax

find\_loop:

get\_char\_by\_inedx bstr, di

mov dh, dl

get\_char\_by\_inedx fstr, ax

cmp dh, dl

je equal

jmp not\_equal

equal:

inc ax

cmp al, bl

je success

jmp return

success:

dec ax

sub di, ax

mov ax, di

jmp exit

not\_equal:

xor ax, ax

jmp return

return:

inc di

loop find\_loop

mov ax, -1

exit:

endm

Удаление подстроки:

remove\_word macro str1, str2

find\_substr str1, str2

cmp al, 0ffh

je remove\_word\_exit

jmp word\_check

word\_check:

push\_registers

xor dx, dx

get\_size str1

mov bx, dx

xor dx, dx

get\_size str2

cmp bx, dx

je continue\_removing

cmp al, 0h

je after\_space

xor dx, dx

get\_size str2

add dl, al

mov bl, dl

get\_size str1

cmp dl, bl

je before\_space

both\_spaces\_1:

dec al

xor ah, ah

get\_char\_by\_index str1, ax

cmp dl, 20h

je both\_spaces\_2

jmp remove\_word\_exit

both\_spaces\_2:

inc al

xor dx, dx

get\_size str2

mov bx, dx

xor ah, ah

add bx, ax

xor dx, dx

get\_char\_by\_index str1, bx

cmp dl, 20h

je increase\_substring\_size

jmp remove\_word\_exit

before\_space:

dec al

xor ah, ah

get\_char\_by\_index str1, ax

cmp dl, 20h

je beforespace\_exists

jmp remove\_word\_exit

beforespace\_exists:

pop\_registers

dec al

push\_registers

jmp increase\_substring\_size

after\_space:

xor dx, dx

xor bx, bx

get\_size str2

mov bx, dx

xor dx, dx

get\_char\_by\_index str1, bx

cmp dl, 20h

je increase\_substring\_size

jmp remove\_word\_exit

increase\_substring\_size:

pop\_registers

inc bl

push\_registers

jmp continue\_removing

continue\_removing:

pop\_registers

get\_size str1

mov cl, dl

sub cl, bl

xor dh, dh

mov ah, bl

mov bx, offset str1

add bx, 2 ;bx -- 0 cell address in the string

add dx, bx

dec dx ;dx -- last cell address in the string

push ax ;ah -- substring length, al -- index of the first occurrence of the substring

mov ah, 0

add bx, ax

mov di, bx

pop ax

push ax

mov al, ah

mov ah, 0

add bx, ax

mov si, bx

xor dh, dh

word\_remove\_loop:

cmp si, dx

ja word\_end\_loop

mov ax, [si]

mov [di], ax

;movsb

inc di

inc si

loop word\_remove\_loop

word\_end\_loop:

pop ax

mov al, ah

mov ah, 0

mov bx, offset str1

add bx, dx

sub bx, ax

inc bx

mov [bx], '$'

mov bx, offset str1

inc bx

get\_size str1

sub dx, ax

mov [bx], dl

remove\_word\_exit:

endm

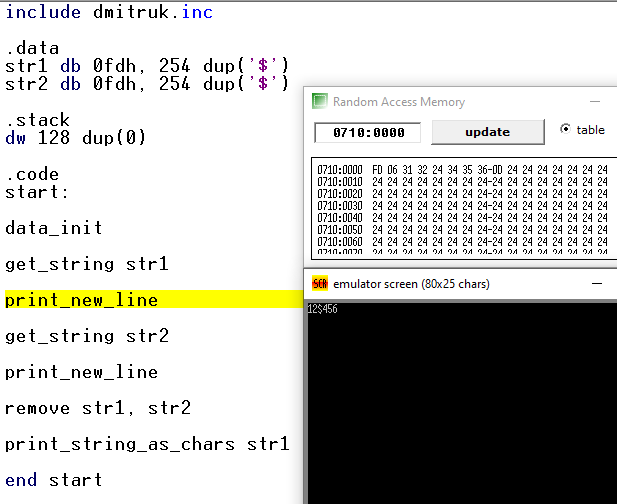


Рисунок 1 – Введена первая строка

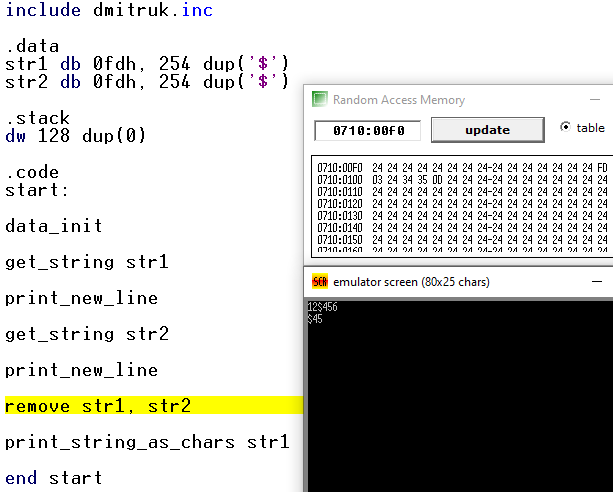


Рисунок 2 – Введена вторая строка

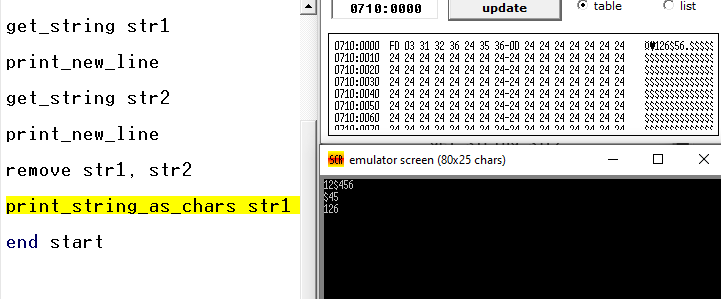


Рисунок 3 – Подстрока удалена

**Выводы:** в результате лабораторной работы с использованием команд обработки символьных данных была написана программа, производящая удаление введенной подстроки из введенной строки.