Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №3

за 2 семестр

По дисциплине: «Компьютерные системы и сети»

Тема: «Макроопределения»

Выполнил:

Студент 2 курса

Группы ПО-6(2)

Осьмушников А.Ю.

Проверил:

Бойко Д.О.

Брест, 2022

Лабораторная работа №3

Макроопределения

Вариант 7

**Задание:**

Изучить состав и средства задания макроопределений.

1. Написать макроопределение, реализующее функцию заданного преподавателем варианта работы: *в исходной строке оставить между словами лишь по одному пробелу, удалив лишние;*
2. Написать программу проверки работоспособности разработанного макроопределения.

**Текст программы:**

.model small

.stack 1000h

.data

stroka db 215 dup ('$')

newstroka db 215 dup ('$')

.code

delspace macro ; //Макроопределение удаления лишних пробелов в строке

mov si, 0 ; //Устанавливаем счетчики на 0

mov di, 0

iter:

mov dl, ‘.’ ;// Устанавливаем точку в регистр dl

cmp [bx + si], dl ; //Проверяем равно ли значение точке

je endf ; //Если равно то переходим в end

mov dl, ‘ ‘ ;//Устанавливаем пробел в регистр dl

cmp [bx + si], dl ; //Проверяем равно ли значение пробелу

je yes ; //Если равно то переходим в yes

jne no ; //Если не равно то переходим в no

no:

mov dl, [bx + si] ; //Устанавливаем значение символа в регистр dl

mov [bp + di], dl ; //Устанавливаем значение dl в копию строки

inc si ; //Увеличиваем счетчик строки

inc di ; //Увеличиваем счетчик копии строки

jmp iter ; //Переходим в iter

yes:

inc si ; //Увеличиваем счетчик строки

jmp iter ; //Переходим в iter

endf:

mov [bp + di], dl ; //Копируем символ из dl в копию строки

endm ; //Конец макроопределения

mov ax, @Data ; //Копируем Data в ax

mov ds, ax ; //Устанавливаем регистр ds на сегмент данных Data

mov bx, offset stroka ; //Устанавливаем в bx исходную строку

mov bp, offset newstroka ; //Устанавливаем bp как копию строки

mov si, 0 ; //Устанавливаем счётчик на 0

read:

mov ah, 01h ; //Устанавливаем код системного вызова на 01h

int 21h ; //Производим системный вызов

mov [bx + si], al ; //Считываем строку в bx

cmp al, ‘.’ ; //Сравниваем конец с символом точки

je start ; //Если конец равен точке то переходим в start

inc si ; //Увеличиваем si на 1

jmp read ; //Переходим в read если ввод некорректен

start:

delspace ; //Вызываем макрокоманду del\_space

mov ah, 02h ; //Устанавливаем системный вызов вывода

mov dl, 13 ; //Задаем значение для вывода

int 21h ; //Производим системный вызов

mov dl, 10 ; //Задаем значение для вывода

int 21h ; //Производим системный вызов

mov si, 0 ; //Задаем счётчик si на 0

mov cl, ‘.’ ; //Устанавливаем значение символа точки в регистр cl

jmp write ; //Переходим в write

write:

cmp [bp + si], cl ; //Сравниваем символ с точкой

je ending ; //Если точка то переходим в ending

mov ah, 02h ; //Устанавливаем системный вызов вывода

mov dl, [bp + si] ; //Задаем значение для вывода

int 21h ; //Производим системный вызов

inc si ; //Увеличиваем счётчик si на 1

jmp write ; //Переходим в write

ending:

mov ax, 4C00 ; //Устанавливаем системный вызов выхода из программы

int 21h ; //Производим системный вызов

end

**Результаты тестирования программы:**

**Вывод:** в ходе данной лабораторной работы написали на ассемблере программу обработки текста.