Министерство РФ по связи и информатизации

Поволжская государственная академия телекоммуникаций

и информатики

Кафедра «ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА»

КОНТРОЛЬНАЯ РАБОТА

ПО ПРЕДМЕТУ

«Организация ЭВМ и систем»

Самара

2009 г.

Программа копирования строки в обратном порядке

Составить программу копирования строки из 15 символов в рядом расположенную область памяти, но в обратном порядке.

Для этого мы создаём две строки. Одна выполняет роль источника данных, а другая – приёмник. Затем мы определяем адрес источника и получателя с помощью команды Lea, которая является альтернативой оператору ассемблера offset. В отличие от offset команда lea допускает индексацию операнда, что позволяет более гибко организовать адресацию операндов. Так как нам нужно заполнить память в обратном порядке, то мы сразу смещаем адрес приёмника DI на 15 байтов. Организовываем цикл и посредством промежуточного регистра AL заносим наши символы в регистр DI, уменьшая, с каждым шагом, байты записи. После того, как счётчик CX уменьшится до 0, программа завершит свою работу

ЛИСТИНГ ЗАДАЧИ «Программирование строки в обратном порядке»

Задача 17

Data Segment ; открыть сегмент данных

text1 DB 'My name Valerij', 13, 10, '$'

text2 DB 25 DUP(?) ; строка получатель

Data ENDS ; Закрыть сегмент данных

;-----------------------------------------------

STK Segment ; открыть сегмент данных

DB 100 DUP(?) ; зарезервировать под стек 100 ячеек памяти

STK ENDS ; закрыть сегмент стека

;---------------------------------------------------

ASSUME SS:STK, CS:CODE, DS:DATA ;назначить сегментные регистры

;-----------------------------------------------------

CODE Segment ; открыть сегмент кода

START: mov AX, DATA ; загрузка сегментного регистра DS

mov DS, AX ; регистра DS

lea SI, TEXT1 ; начальный адрес строки-источника занести в SI

lea DI, TEXT2 ; начальный адрес строки получателя занести в DI

mov CX, 15 ; переслать 15 байтов

Metka1: mov AL, [SI] ; выполнить пересылку

mov [DI + 15], AL ; через промежуточный регистр AL

INC SI ; увеличить адрес источника на 1

DEC DI ; уменьшить адрес источника на 1

LOOP Metka1 ; (CX)-1, повторить цикл пока CX не равно 0

mov AL, 0 ; завершить программу через

mov AH, 4Ch ; системную функцию возврата

int 21h ; в DOS

CODE ENDS ; закрыть сегмент кода

END START ; закрыть программу

Ввод строки

**abcdefghijklmno**

CX=15

Выполняем перемещение:

**mov AL, [SI]**

mov [DI], AL

INC SI

DEC DI

CX=0

Программа копирования строки на прямую

Составить программу производящую копирование строки из 15 символов в расположенную рядом область памяти.

Задача, практически идентичная предыдущей. За исключением способа пересылки строки. В данном случее нам уже не требуется вести запись в обратном порядке. Также, создаём две строки. Одна выполняет роль источника данных, а другая – приёмник. Затем мы определяем адрес источника и получателя с помощью команды Lea. Приёмник DI на 15 байтов больше не смещаем.

Организовываем цикл и посредством промежуточного регистра AL заносим наши символы в регистр DI. При каждом шаге байты будем увеличивать в регистре DI на один. После того, как счётчик CX уменьшится до 0, программа завершит свою работу.

ЛИСТИНГ ЗАДАЧИ «Программа копирования строки на прямую»

Задача 1

Data Segment ; открыть сегмент данных

text1 DB 'abcdefghijklmno', 13, 10, '$'

text2 DB 25 DUP(?) ; строка получатель

Data ENDS ; Закрыть сегмент данных

;-----------------------------------------------

STK Segment ; открыть сегмент данных

DB 100h DUP(?) ; зарезервировать под стек 100 ячеек памяти

STK ENDS ; закрыть сегмент стека

;---------------------------------------------------

ASSUME SS:STK, CS:CODE, DS:DATA ;назначить сегментные регистры

;-----------------------------------------------------

CODE Segment ; открыть сегмент кода

START: mov AX, DATA ; загрузка сегментного регистра DS

mov DS, AX ; регистра DS

lea SI, TEXT1 ; начальный адрес строки-источника занести в SI

lea DI, TEXT2 ; начальный адрес строки получателя занести в DI

mov CX, 15 ; переслать 15 байтов

Metka1: mov AL, [SI] ; выполнить пересылку

mov [DI], AL ; через промежуточный регистр AL

INC SI ; увеличить адрес источника на 1

INC DI ; увеличить адрес приёмника на 1

LOOP Metka1 ;(CX)-1 повторить цикл пока CX не равно 0

mov AL, 0 ; завершить программу через

mov AH, 4Ch ; системную функцию возврата

int 21h ; в DOS

CODE ENDS ; закрыть сегмент кода

END START ; закрыть программу

Ввод строки

**abcdefghijklmno**

CX=15

Выполняем перемещение:

**mov AL, [SI]**

mov [DI], AL

INC SI

INC DI

CX=0