Министерство образования Республики Беларусь

Учреждение Образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Лабораторная работа №7

по дисциплине «Программирование на языке ассемблера»

на тему «Загрузка и выполнение программ. Работа с памятью»

вариант №7

|  |  |  |
| --- | --- | --- |
| Выполнил  студент гр. 250501  Лукьянов Е.О. |  | Проверил  Туровец Н.О. |

Минск 2023

**Цель работы:** Ознакомиться с загрузкой и выполнением программ, рассмотреть работу с памятью.

**Теоретические сведения**

Для выполнения работы требуется рассмотреть следующие элементы языка ассемблера и операционной системы:

1. Управление памятью.

При запуске программы в DOS, ей выделяется все доступная память, поэтому доступно использование, например, памяти за концом программы практически до конца сегмента кода. Однако для загрузки других программ из текущей потребуется свободная память и наиболее простой вариант ее получения – сократить выделенный текущей программе блок памяти до минимума с помощью функции DOS 4Ah.

-- Функция DOS 4Ah (INT 21h) – изменить размер блока памяти:

Ввод: АН = 4Ah,

ВХ = новый размер в 16-байтных параграфах,

ES = сегментный адрес модифицируемого блока.

Вывод: если CF = 1, то есть ошибки (в AX – код ошибки):

-- 07 – блоки управления памятью разрушены,

-- 08 – не хватает памяти (при увеличении, в ВХ = максимальный размер, доступный для этого блока),

-- 09 – ES содержит неверный адрес.

Если CF = 0, то операция выполнена успешно.

Также доступно выделение и удаление дополнительных блоков памяти: -- Функция DOS 48h (INT 21h) – выделить блок памяти:

Ввод: АН = 48h ВХ = размер блока в 16-байтных параграфах.

Эта функция с ВХ = FFFFh используется для определения размера самого большого доступного блока памяти.

Вывод: если CF = 1, то есть ошибки (в AX – код ошибки):

-- 07 – блоки управления памятью разрушены,

-- 08 – не хватает памяти (ВХ = размер максимального доступного блока).

Если CF = 0, то операция выполнена успешно,

АХ = сегментный адрес выделенного блока.

-- Функция DOS 49h (INT 21h) – освободить блок памяти:

Ввод: АН = 49h

ES = сегментный адрес освобождаемого блока

Вывод: если CF = 1, то есть ошибки (в AX – код ошибки):

-- 07 – блоки управления памятью разрушены,

-- 09 – ES содержит неверный адрес.

Если CF = 0, то операция выполнена успешно.

1. Загрузка и выполнение программ.

Для загрузки и выполнения программ требуется использовать функцию

DOS 4Bh (INT 21h) – загрузить и выполнить программу:

Ввод: АН = 4Bh,

AL = подфункции:

AL = 00h — загрузить и выполнить;

AL = 01h — загрузить и не выполнять:

DS:DX – адрес ASCIZ-строки с полным именем программы,

ES:BX – адрес блока параметров ЕРВ:

+00h (слово) – сегментный адрес окружения, которое будет скопировано для нового процесса (или 0, если используется текущее окружение);

+02h (4 байта) – адрес командной строки для нового процесса;

+06h (4 байта) – адрес первого FCB для нового процесса;

+0Ah (4 байта) – адрес второго FCB для нового процесса;

+0Eh (4 байта) – здесь будет записан SS:SP нового процесса после его завершения (только для AL = 01);

+12h (4 байта) – здесь будет записан CS:IP (точка входа) нового процесса после его завершения (только для AL = 01).

AL = 03h – загрузить как оверлей:

DS:DX — адрес ASCIZ-строки с полным именем программы,

ES:BX — адрес блока параметров: +00h (слово) – сегментный адрес для загрузки оверлея,

+02h (слово) – число, которое будет использовано в командах, использующих непосредственные сегментные адреса (обычно то же самое число, что и в предыдущем поле или 0 для com-файлов).

AL = 05h – подготовиться к выполнению (DOS 5.0+):

DS:DX – адрес следующей структуры:

+00h(слово) – 00h,

+02h (слово) :

бит 0 – exe-программа,

бит 1 – программа-оверлей,

+04h (4 байта) – адрес ASCIZ-строки с именем новой программы;

+08h (слово) – сегментный адрес PSP новой программы;

+0Ah (4 байта) – точка входа новой программы;

+0Eh (4 байта) – размер программы, включая PSP.

Вывод: Если CF = 1, то произошла ошибка (в АХ = код ошибки):

-- 02h – файл не найден,

-- 05h – доступ к файлу запрещен,

-- 08h – не хватает памяти,

-- 0Ah – неправильное окружение,

-- 0Bh – неправильный формат.

Если CF = 0, то операция успешно выполнена:

ВХ и DX изменены.

Особенности: Для подфункций 00 и 01 требуется, чтобы было достаточно свободной памяти для загрузки программы, поэтому com-программы должны воспользоваться функцией DOS 4Ah для уменьшения отведенного им блока памяти до минимально необходимого. При вызове подфункции 03, DOS загружает оверлей в память, выделенную текущим процессом, так что exe-программы должны убедиться, что ее достаточно. Эта функция игнорирует расширение файла и различает exe- и com-файлы по первым двум байтам заголовка («MZ» для exe-файлов).

1. Оверлейные модули.

Оверлей – это часть исполняемой программы (обычно процедура, хотя это может быть полностью самостоятельная программа со своими сегментами данных и стека), которая по мере необходимости загружается в определенную область памяти. Различные оверлейные модули могут загружаться в одно и то же место, перекрывая предыдущий код, что позволяет экономить память, но снижает быстродействие программы при частых загрузках.

**Код программы (.exe)**

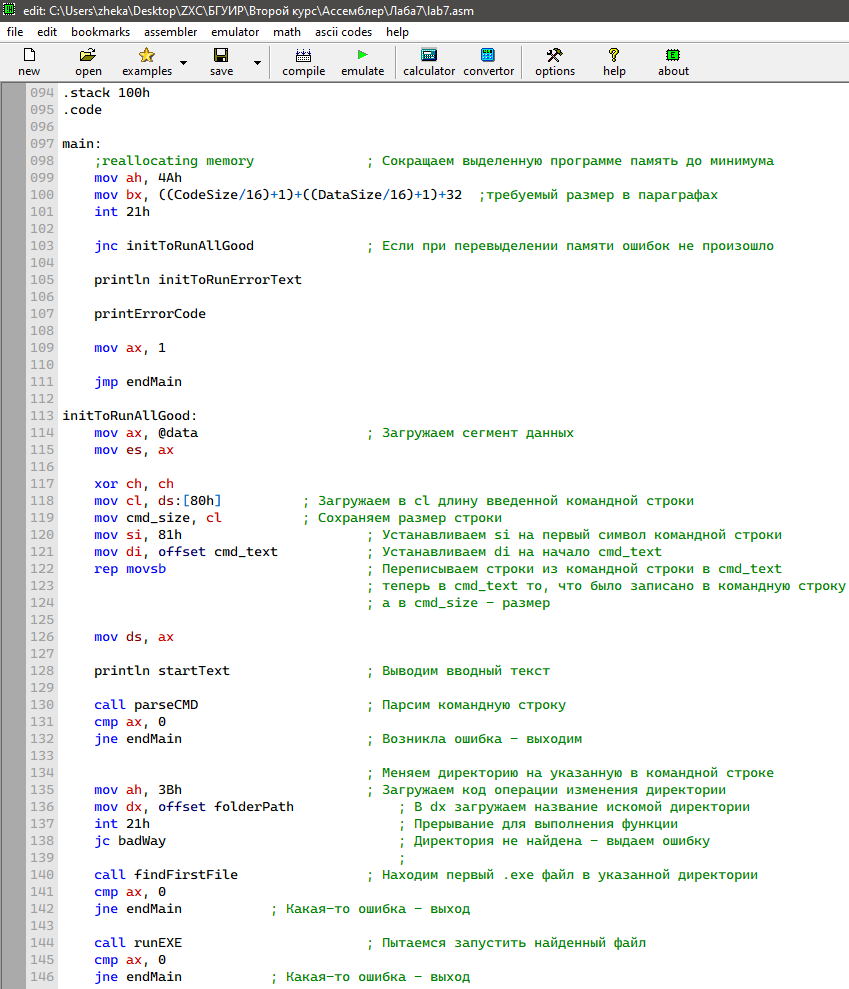
****

Рисунок 1 – Код программы

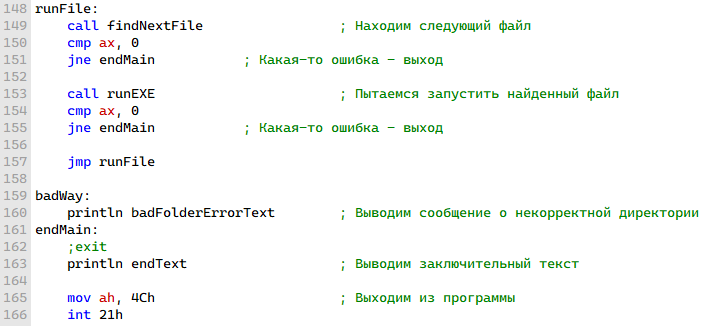
****

Рисунок 2 – Код программы

**Вывод программы**

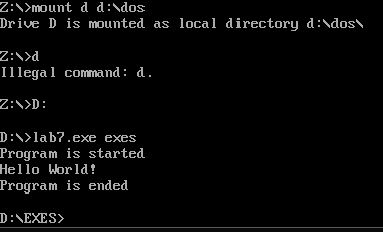
****

Рисунок 3 – Результат работы программы